AN INTEGRATED PRODUCT, PROCESS AND RESOURCE MODELING TECHNIQUE FOR COST ESTIMATION: A REMOTE LASER WELDING CASE.

By

Ken Asare

A thesis submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy in Engineering

Warwick Manufacturing Group University of Warwick

United Kingdom, CV4 7AL

2017

DEDICATION

This work is dedicated to my beloved wife, Mina, for her endless support, my children for their understanding and my Mum and Dad who toiled, cheered and waited for this day.

DECLARATION

This thesis is presented in accordance with the regulations for the degree of Doctor of Philosophy. It has been written by myself and has not been submitted anywhere else. The work in this thesis has been undertaken by me except where otherwise stated.

ACKNOWLEDGEMENTS

I am thankful to God Almighty for the grace He offered me to be able to accomplish this research study.

Words cannot express how grateful I am to Professor Dariusz Ceglarek, Dr Neil Davis and Professor Kwabena Agyapong-Kodua whose supervision, direction and concern enabled me to successfully complete this work.

My gratitude also goes to all the past and current members of the Digital Lifecycle Management (DLM) lab of Warwick Manufacturing Group (WMG), University of Warwick, particularly Manoj Babu, Erkan Ozkat, Dr Pasquale Franciosa and Dr Aniruddha Gupta who made my stay at the DLM lab an enjoyable one.

My appreciation goes to EU-FP7 FoF-ICT-2011.7.4: "Remote Laser Welding System Navigator for Eco &Resilient Automotive Factories" (#285051) for the support received for my research projects. I would like to appreciate Mr Charles Marine for his sincere opinion and industrial point of view and support towards my research.

I also acknowledge the financial support received from WMG, University of Warwick throughout my doctoral research.

I would also like to express my sincere appreciation to my wife, Mina and children: Jesse, Eliane, Christelle and Emmanuel for their encouragement, support and cooperation with me during my time of study. To my siblings, extended families and in-laws, I thank you all for your encouragement, understanding and prayer support.

Special thanks to LCI Coventry members for their prayer support and to my IT consultant, Mr Jude Fletcher for technically supporting, bless you all.

ABSTRACT

Current advances in manufacturing systems engineering have led to the development of tools and techniques that support enterprises to remain in business, be competitive and at the same time deliver products that meet the needs of their customers. Some commonly identified tools and techniques observed in literature today are: Cost Modelling techniques such as Activity Based Costing, Parametric, Expert judgment, Analogy, etc; System Modelling Techniques such as Knowledge Based Engineering; Enterprise Modelling and Behavioural Modelling tools using IT systems (Static, Dynamic and DES models) to help understand and represent systems.

Although current tools and techniques have demonstrated significant benefits to enterprises, the following challenges are observed in literature; (i) the lack of an integrated engineering and cost modelling technique during early design stages, (ii) lack of a systematic execution of engineering models into existing cost modellers to extend its capabilities to include new processes is not well documented. Which means that currently, most cost modellers are only capable of estimating cost with existing inbuilt models which depend on domain experts modification, (iii) engineering cost estimation tools that only generates manufacturing cost of products which does not include design and installation cost for new products that requires new processes. Addressing these challenges may lead to the development of dedicated and more integrated tools for useful collaborative analyses during early design phases. Also, this may lead to the development and improvement of digital modelling tools that represent actual conditions of production systems and facilities. Furthermore, this may also support organisations with the capturing of engineering knowledge to help understand processes as well as to have a better overview of capital investments.

This research proposes a *Product-Process-Resource (PPR) Cost Estimation Framework* to satisfy the above challenges. The proposed framework is developed through three interlinked techniques.

The first technique is a *"product-process-resource modelling technique for capturing engineering knowledge and cost values"* addresses challenge (i) above. This technique uses data modelling approach for capturing engineering knowledge and extracts cost information for assessing product (P), process (P) and resources (R) design cost during early design stages. Engineering knowledge in this context refers to an understanding of engineering processes and resources that are consumed or expected to be consumed to realize a particular product or features on a product. This technique makes use of business process modelling notations (BPMN) to illustrate the integration of process and resources. Furthermore, a computer representation of the process with its workstation are generated in an Extensible Markup Language (XML) format, which are both human-readable and machine-readable to be used in the next stage of the proposed methodology. Also, a PPR Design Cost Calculator is developed for capturing cost of design using standard cost accounting algorithms. The calculator enables engineers to visualise

cost values of product, process and resource design and changing cost parameters to see its effect on the total cost. This technique is based on the assumption that product features can be associated with process capabilities which then can be mapped onto resource competencies and capacities.

The second technique is a "*technique for extending cost modeller capabilities to include a new process for cost assessment"* addresses challenge (ii) above. This solution is based on identifying cost modeller requirements and then developing and implementing compatible product, process and resource models to extend the cost modeller's capabilities. These models as discussed in the first approach becomes input for extending the capabilities of a cost modeller for costing a specific feature on a product 3D model.

Finally, challenge (iii) is addressed by introducing a "*technique for integrating P-P-R-Production cost values to support engineering decisions"*. This is a unique technique that integrates PPR design cost models, installation cost model and production cost database to generate visible cost values as a cost summary. The cost summary contains cost algorithms and equations, modelled to reflect the effect of time, rate, annual production volume, material and batch size changes to the total cost.

The proposed PPR Cost Estimation Framework has been verified and validated with an industrial case study of an automotive sheet metal door assembly process, a novel Remote Laser Welding (RLW) case application for its rigorousness and future industrial applicability.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF PUBLICATIONS

Asare, K. B, Agyapong-Kodua, K., & Ceglarek, D. J. 2014. Digital Modelling Methodology for Effective Cost Assessment. Procedia CIRP, 17, 744-749.

K. B. Asare, M. Shang, D. Ceglarek, K. Agyapong-Kodua, 2017. Submitted to Research in Engineering Design. *A Product, Process, Resource and Cost integrated methodology for effective system design*. Under review.

K. B. Asare, M. Shang, D. Ceglarek, K. Agyapong-Kodua, 2017. Submitted to International Journal of Computer Integrated Manufacturing. Life Cycle Design, Production Modelling, Simulation Manufacturing Systems Engineering.

K. B. Asare, D. Ceglarek, N. Davis, K. Agyapong-Kodua and C. Marine, 2017. '*A Product-Process-Resource Modelling Technique for Capturing Engineering Knowledge and Cost Values*'. To be submitted to International Journal of Industrial Engineering Computations.

K.B Asare, D. Ceglarek, N. Davis, K. Agyapong-Kodua, 2017. '*A Technique for Extending Cost Modeller Capabilities to Include a New Process For Cost Assessment.*' To be submitted to International Journal of Computer Integrated Manufacturing.

K. Asare, K. Agyapong-Kodua and N. Davis, 2017. '*An Integrated Product-Process-Resource-Production Cost Estimation Technique'*. To be submitted to Factory 2050 Journal of Manufacturing.

ABBREVIATIONS

ABC:	Activity Based Costing
AI:	Artificial Intelligence
CAD:	Computer Aided Design
CAE:	Computer-Aided Engineering
CAM:	Computer Aided Manufacturing
CER:	Cost Estimating Relationship
CIMOSA:	Open System Architecture for CIM
CMWB:	Cost Model Workbench
COQ:	Cost of Quality
CSL:	Cost Scripting Language
D*f*A:	Design for Assembly
D*f*M:	Design for Manufacturing
DSM:	Decision Support Model
EA:	Enterprise architecture
EE:	Enterprise Engineering
EI:	Enterprise Integration
EM:	Enterprise modelling
GERAM:	Generalised Enterprise Reference Architecture and Methodology
ICAD:	Intelligent Computer Aided Design
IGES:	International Graphics Exchange Standard
KBE:	Knowledge Based Engineering
KPI:	Key Performance Indicator
ME:	Manufacturing Enterprises
MIKE:	Model-based and Incremental Knowledge Engineering
MOKA:	Methodology and tools Oriented to knowledge based Applications
NPD:	New Product Development
NPI:	New Product Introduction
PERA:	The Purdue Enterprise Reference Architecture
PPR:	Product-Process-Resource
PPRC:	Product-Process-Resource-Cost

RA:	Reference Architecture
RLW:	Remote Laser Welding
RSW:	Resistance Spot Welding
SPR:	Self-Piercing Riveting
STEP:	Standard for the Exchange of Product Model Data
TOGAF:	The Open Group Architectural Framework
VFF:	Virtual Factory Framework’s
VPE:	Virtual Production Environment
XML:	Extensible Markup Language

CHAPTER 1

INTRODUCTION

1.1 Introduction

Manufacturing Enterprises (MEs) are currently facing extreme levels of competition due to complex customer demands/changes and other challenges such as technological, social, environmental, legal and political issues. To remain competitive, there is the need for MEs to deploy strategies and techniques that will enable them to become flexible in their design, manufacturing and service operations. It is however estimated that about 70% of manufacturing cost is determined at the design stage, with very limited production information (Jin, 2012). Most decisions which are cost sensitive are taken at early stages of the design phase and it is, therefore, necessary to support designers to understand the cost implication of their engineering decisions (Agyapong-Kodua et al., 2012). Pragmatic techniques are used by many manufacturing companies today to improve their manufacturing operations by reducing the time taken to carry out activities, processes, projects or programs from conception through to delivery to the customer.

To survive in an ever-changing economic environment, manufacturing enterprises are increasing the use of modelling techniques and simulation tools for product, process and resources, mainly focusing on lifecycle simulations and design for product variation to achieve high-quality product and robust processes (Maropoulos and Ceglarek, 2010). The main objectives for using these tools and techniques are to understand the system's dynamic in order to reduce the total process cycle times for sections of departments or for the entire organization. Enterprises have adopted the ubiquitous technology of using simulation tools and models in today's cutthroat environment. Generally, simulations deployed by MEs are to encourage parallel processing during the product development stage; shorten the product development cycle times; and to avoid the cost of physical prototype building (Lee et al., 2011). The need for modern manufacturing systems to be able to meet several challenges and at the same time satisfy customer requirements and constraints that change over time is of a great importance. As simulations are carried out, products, processes or resources may undergo various engineering changes critical to enterprise's success.

The manufacturing systems life cycle approach looks at the manufacturing system and the entire factory as a product characterized by several stages: design stage, implementation stage, operation stage and subsequent re-design/reconfiguration of the manufacturing system (ElMaraghy, 2005); (Westkämper et al., 2005) (Westkämper, 2007). However, complexities in manufacturing systems trigged the idea of analytical methods (Chung and Synder, 1999), (Heragu and Kusiak, 1988), (Hamann and Vernadat, 1992), (Kim and Kim, 2000) which are currently employed by many MEs ((Gershwin, 1994); (Matta et al., 2005)). As these complexities in manufacturing systems are ever increasing, the use of information technology (IT) is employed to handle some of the complexities and at the same time reduce time and cost for manufacturing innovation and productivity enhancement by supporting several stages in the design, development, production and operation of novel manufacturing systems (Alexopoulos et al., 2011), (D'Addona and Teti, 2011). It is always expected in every organization that design, manufacturing departments and cost knowledge work together seamlessly but in reality, there is a gap between them. This is supported by Agyapong-Kodua et al. (2014) research on digital modelling methodology concluded that the implications of such gap result in time consuming and unnecessary process activities that generate extra expenses due to several levels of iterations required to reach optimal decisions.

Since most engineering decisions which are cost sensitive are taken at early stages of the design phase, it is necessary to support designers to understand the cost implication of their engineering decisions (Agyapong-Kodua et al. 2012; Jarratt et al. 2011). Achieving this is not trivial because there are a number of competitive key performance indicators that designers will have to control to reach optimal design solutions (Shrouti et al. 2013). To manage this challenge, proponents of cost engineering and accounting (Akintoye and Fitzgerald 2000; Caputo and Pelagagge 2008a; Cavalieri et al. 2004; Curran et al. 2004a) have recommended a number of approaches for cost estimation of projects, lifecycle analysis, technology down selection and assessment of economic viability of engineering projects. A review of these bodies of literature, however, shows that due to the inherent complexities and dynamic changes in product, process and resource requirements, it is fairly difficult to estimate, predict, control and monitor cost consumption appropriately. It was also noted that traditional cost accounting practices are best deployed to manage and control cost during operational stages of manufacturing systems but less helpful during early stages of product, process or resource systems design. Coupled with this, traditional cost accounting practices have not been kept up with the advances in design and manufacturing technologies (Agyapong-Kodua et al. 2012c). Current cost accounting techniques may be able to provide ‘static cost’ impressions when fed with suitable information but limited in predicting cost as a result of frequent engineering changes of dimensions, materials, tolerances, shapes and so forth (Asiedu et al. 2000).

Cost is an important key performance indicator (KPI) amongst others which need to be considered at the very early stages of product design. Carter and Baker (2002) argued that cost should be incorporated in the conceptual phase of a product development. According to their research, the scale of cost increases by almost a factor of 10 when changes have to be made at the next level of the product lifecycle. This, however, is to emphasize the major benefit of saving by means of increasing an organization’s profit margin through controlling cost at the concept and design phases of a new product introduction (NPI) process. In support of this, Agyapong-Kodua et al. (2012) inferred that designers are to be supported to understand the cost implication of their engineering decisions early enough. More critically, Shrouti et al. (2013), mentioned that this is not trivial as there are other KPIs designers will have to control to reach optimal design solutions of which cost may not necessarily be one. To overcome this challenge, experts in cost engineering and accounting recommended approaches such as: cost estimation of projects, lifecycle analysis, technology down selection and assessment of economic viability of engineering projects (Akintoye and Fitzgerald (2000); Caputo and Pelagagge (2008); Cavalieri et al. (2004); Curran et al. (2004).

A new product that can be realized by existing processes and existing resources may come with little complexities for manufacturing planning and cost estimation purposes. Likewise, developing a new technology (process and resources) to realize existing product to manufacture in a more cost efficient way may not also exhibit too much complexity. However, the challenge is this, where a new product and a new process are required, what systematic approach does MEs take to ensure that all necessary knowledge requirements are identified and captured in a way that ensures repeatability of the method? In many instances, estimating the cost of a product or a process is done by cost estimators using their expert judgement which may result in overestimating or underestimating. Other means of estimating today is the use of cost estimation tools. The negative side of the latter is that, cost estimation tools has predefined processes and resources database which may not necessarily reflect reality, and hence, although that estimate may be good, it may not be suitable. There is, therefore, the need to develop a generic system that is capable of capturing, use, storing and managing the knowledge of creating a process for a new product that generates cost information that is repeatable for other applications. This will support and reduce engineering decision making time by bridging the gap between design and manufacturing through the introduction of cost as a key performance indicator (KPI) for both. This is achieved by integrating product, process and resource models for a manufacturing system in a way that cost information is generated early at various design stages. Also, to ensure that the cost values are available to both design and manufacturing as geometric features and manufacturing feature undergo various iterations. This will give designers options to choose alternative process or resource where there is an engineering change and at the same time identify the cost implications of the engineering change.

To satisfy the above challenges, this research proposes a modelling framework called Product-Process-Resource (PPR) Cost Estimation Framework. The proposed framework is developed through three interlinked approaches.

The first approach is a data modelling technique that captures engineering knowledge and extracts cost information for estimating product (P), process (P) and resources (R) design cost during early design stages. Engineering knowledge in this context refers to an understanding of engineering processes and resources that are consumed or expected to be consumed to realize a particular product or features on a product. This technique makes use of business process modelling notations to illustrate the integration of process and resources. The process is divided into workstations where the integration is further carried out at the workstation levels, to show detailed activities of resources at each workstation. Furthermore, a computer representation of each workstation is generated Extensible Markup Language (XML), which are both human-readable and machine-readable to be used in the next stage of the proposed methodology. Also, a PPR Design Cost Calculator is developed that captures the times and rates based on a unique cost algorithm created. The calculator enables engineers to visualize cost values of product, process and resource design and changing cost parameters to see its effect on the total cost. This technique is based on the assumption that product features can be associated with process capabilities which then can be mapped onto resource competencies and capacities.

Secondly, a systematic approach of executing engineering models into a cost modeller to extend its existing cost modeller capabilities is addressed. This solution is based on identifying cost modeller requirements and implementing compatible product, process and resource models that satisfy the requirements. These models as discussed in the first approach become inputs for extending the capabilities of a cost modeller for estimating the cost of a specific feature on a 3D CAD model of a product.

Thirdly, an integrated P-P-R-Production cost estimation technique was introduced. This is a unique technique that integrates PPR design cost database and production cost database and makes cost values visible to both design engineers and manufacturing in a cost summary. The cost summary contains cost algorithms and equations that are capable of estimating the effect of engineering changes on product design, process design, resource design and the production process costs.

The proposed PPR Cost Estimation Framework have been validated with an industrial case studies of an automotive sheet metal door assembly process, a novel Remote Laser Welding (RLW) case application for its rigorousness and future industrial applicability. Results obtained from the case application were compared with a state-of-the-art method to shows that the proposed framework is capable of generating systematic engineering knowledge capturing during early design stages for estimating cost and for generating instant cost values based on engineering scenarios to support engineering decisions.

1.2 Motivation

Based on reviews of current literature on cost estimation, factory system modelling, knowledge based engineering amongst others in design and manufacturing domain, cost is considered as a very important performance indicator and an ineffective cost decision can cause an organization to be non-competitive and insolvent due to the lack of process understanding. The review further confirmed the relevance of an integrated product, process and resource modelling concept. However, the review also shows that there is the:

(i) Lack of an integrated engineering system modelling and digital cost modelling techniques

Major design and engineering decisions bother on cost but current generation industrial practice dissociates critical engineering activities from cost modelling. Although there is the drive towards integrating these two fields, manufacturing industries are yet to fully benefit from this. In some advanced manufacturing industries, initial CAD models are generated by product designers. Product Designers typically describe the functions, materials specifications and dimensions of products. These details are then used by process engineers to define process sequences, facilities and hence estimate the cost required to realize the product. This initial estimate is based on metrics such as run time per day, tooling and facilities investment required, floor space, labour, material, and job per hour. The manufacturing feasibility is usually checked by the process designer and when there are potential issues with the manufacturability of the product, a change request is sent to the product designers. This cycle could repeat itself for a long time until a suitable solution is obtained. Although digital modelling tools have been developed to help with this process, these tools do not provide adequate support for direct realization of cost implication of design or engineering decisions. Hence these tools have only mimicked the traditional approach to product modelling. In reality, manufacturing, design and cost knowledge are currently isolated although required to complement each other. The implication of such separation can be time consuming and expensive as it may require several levels of iteration to reach improved decisions. The direct integration of engineering decisions with cost will:

1. Reduce product design life cycles and improve cost effectiveness of manufacturing industries
2. Allow the economic implication of alternative product designs to be assessed
3. Experiment the economic effect of introducing different features on products
4. Check alternative processes (as well as resources) and their suitability to meet product requirements

Based on the above points, there is the need for methods and techniques which will define a modelling approach that allows the assessment of cost implications of engineering decisions.

(ii) Lack of full representation of real instances of manufacturing processes for product flow

Current best digital modelling tools do not necessarily recognize actual work loading conditions of production systems, and therefore cost estimation based on these models may not fully represent business conditions of the customized manufacturing system. This is so because best process modelling techniques in support of cost modelling assume single flow dedicated manufacturing systems and underestimate the implication of multiple flows, resource sharing, product mix, rework and other manufacturing dependent failures. This assumption is challenging when real manufacturing instances are to be considered. This is because real manufacturing industries are characterized by complex interrelated systems of processes which may be causally or temporarily related and need to be resourced with technical and human systems. As a result, waiting times, inventory sources, resource availability, actual cycle times may need to be considered in any model geared towards cost modelling. These systems need to be properly coordinated and controlled to allow a variety of products to be realized through them over varying periods of time. Current approaches to the identification and modelling of cost have not been adequate because most of the cost modelling techniques do not encode time dependencies related to product flows, controls, process instants and time dependent causal effects. This implies that analysis of alternative flows of product (volume and mixes) through a shared process cannot accurately be modelled by current generation cost modelling techniques. There is, therefore, the need for research into suitable representations of the dynamic instances of manufacturing systems and how they can be reflected in cost models so that in the product design phase, the behaviour of such systems can be used to support actual cost models.

(iii) Lack of a comprehensive cost modelling tools for developing a new process options to support engineering decisions

Remote laser welding (RLW) is a novel joining process competitive with conventional processes such as resistance spot welding (RSW) and self-piercing riveting (SPR). Literature shows that RLW has potential to be five times faster and occupies 50% less floor space when compared with RSW (Agyapong-Kodua et al., 2014). RLW utilizes 80% fewer robots, less tooling stations and consumes 10% less operation cost (Mori et al., 2010). Energy usage is reported to be reduced by 20% when compared with RSW (Um and Stroud, 2013). Due to these reported benefits, many automotive industries are investing in RLW technologies. The introduction of RLW technologies leads to geometrical changes in product designs. This means products that can be conveniently assembled through RSW and self piercing riveting (SPR) will need modifications for them to be realized through RLW technologies. Initial research focused on the development of systems models (Colledani et al., 2013, Colledani et al., 2014), process details (Shrouti et al., 2013) and monitoring activities (Prakash et al., 2009). Whilst realizing the technical objectives, cost models were developed to justify the economic benefits of RLW compared with other assembly technologies (RSW and SPR) (Matsuda et al., 2012). Other achievements and reported findings include RLW process optimization (laser beam parameters selection, fixture layout optimization), RLW process control (in-line monitoring and control) and RLW Eco-Efficiency evaluation (Prakash et al., 2009). The aim of these research activities was to provide a toolkit to facilitate process planning, design, implementation and optimization of RLW process for Body-In-White Sheet metal joining (Matsuda et al., 2012). Despite earlier achievements, there is need to develop models which will integrate the existing simulation technologies already achieved through the FP7 RLW Navigator project (Matsuda et al., 2012) to enable effectively holistic decision making.

One major objective of the integration process is to be able to assess the cost implication of engineering decisions associated with new features of the RLW-related products. Furthermore, the cost implications of RLW process on product lifecycle from design to service have not been properly investigated and understood. This cost analysis can be applied as a benchmark for investment justification. Therefore, there is a need to develop a cost modelling workbench specifically for RLW process, which can encapsulate the uniqueness of the RLW process features.

As a results of the identified gaps in literature, this research work addresses *(i)*; by integrating engineering system modelling techniques with cost modelling techniques. Also, some aspect of *(iii)* is covered in this research, where a new process (RLW)is introduced into an existing cost estimation tool (aPriori) as a process option to support engineering decisions. The RLW process was selected for the purpose of this research because it is an emerging and novel fabrication process in the automotive body in white (BIW) industry. Although aPriori is capable of costing fabrication techniques such as resistant spot welding (RSW), MIG welding, etc., RLW process and its associated resources are not currently available in the tool. This work therefore introduces RLW as a novel process within aPriori.

1.3 Research Objectives

1) To review literature in manufacturing systems engineering with emphasis on engineering cost estimation, product and features definition, Knowledge Based Engineering, Enterprise modelling and Engineering Decision Making

2) To gather and model data that represents a manufacturing system

3) To create an integrated cost model that generates cost information during design and production stages

4) To integrate the modelled data with a cost modeller

5) To expand the capabilities of a cost modeller to include a new process routing.

6) To generate cost values using scenarios that show the effect of changes to product, process and resource parameters on cost.

1.4 Research Question

How can product, process and resource be integrated in a way that ensures that engineering knowledge, product and process design costs are captured during early design stages when introducing a new manufacturing technology?

1.5 Research Outline

The outline of this research is shown in Figure 1.1.

Chapter 1 introduces the research, outlining the aims, objective and a background to the research.

Chapter 2 reviews literature in Cost estimation and existing NPI processes looking at research gaps that this research aims at satisfying.

Chapter 3 explains the PPR Cost Estimation Framework, showing the systematic flow of data collection, modelling, execution, cost assessment and cost report generation.

Chapter 4 is a case application of the PPR Cost Estimation Framework on a novel Remote Laser Welding (RLW) technology, showing how a new process can be introduced and made available as a process routing option.

Chapter 5 discusses the results of the case application considering some of the benefits of the methodology is discussed as a means of verifying and validating it to prove the generic use of the approach.

Chapter 6 concludes the research with some limitations of the methodology as well as recommendations and future works.

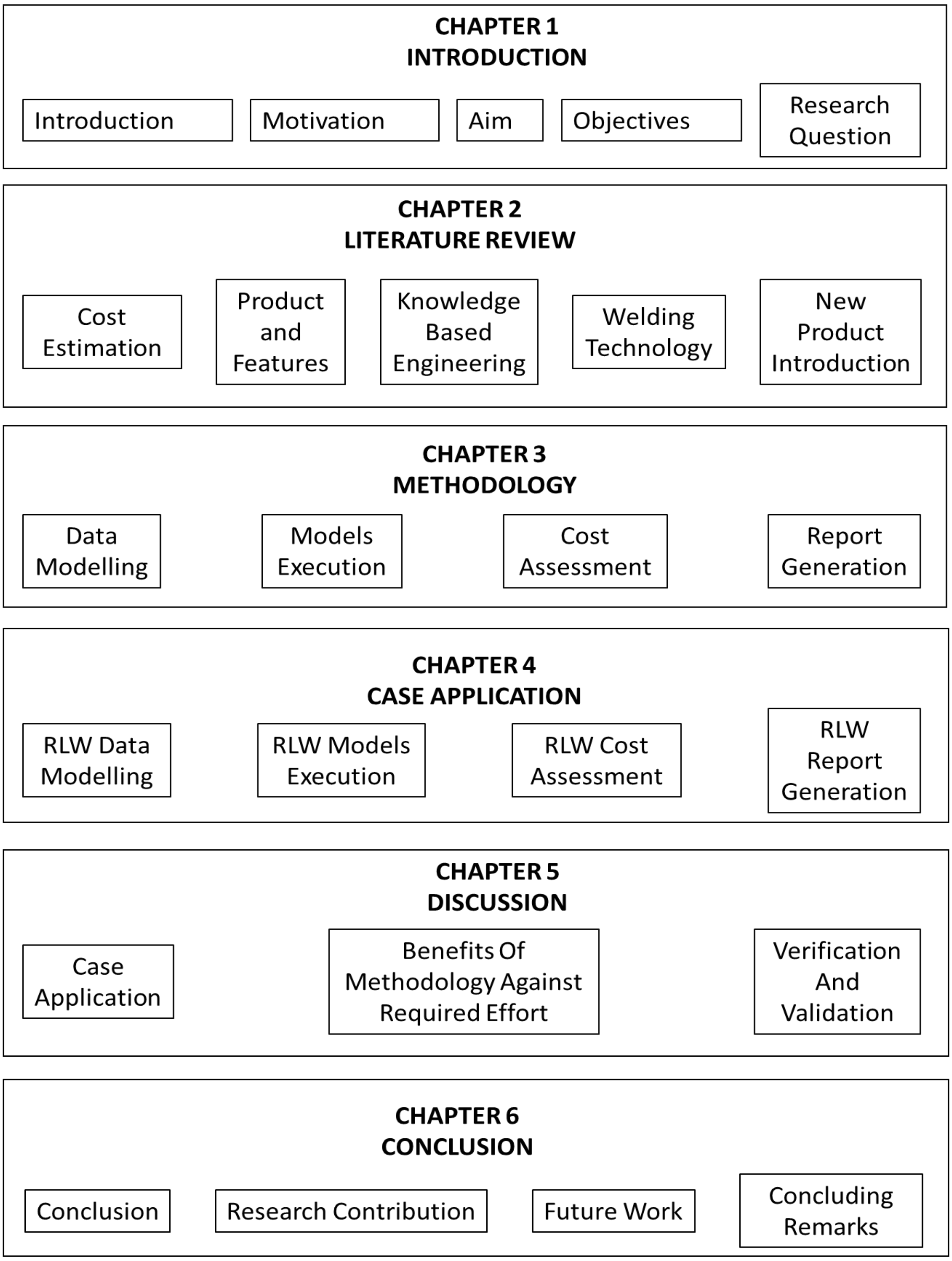


Figure 1.1: Thesis Outline

1.6 Research Scope

Although the proposed methodology in this research may be applicable to other areas, this work is limited to the manufacturing domain as data used was mainly obtained from a laser welding technology project which is a manufacturing process.

Based on the gaps identified in literature in section 1.4, the scope of this is to address *the lack of an integrated engineering system modelling and digital cost modelling techniques* by:

- Modelling and integrating product, process and resources (PPR) of a system using stastic system modelling techniques.
- Reviewing state of the art cost estimation techniques and developing a dynamic cost model that is capable of capturing PPR design cost during early stage design.
- Implementing the developed PPR models into a commercial tool and integrating it with the developed cost model.

CHAPTER 2

LITERATURE REVIEW

This chapter presents a literature review of current work done in the field of manufacturing systems engineering. The topics covered in this section are state of the art approach used in the research domain relevant to this thesis. Seven main sections this chapter which are Section 2.1 introduces engineering cost, Section 2.2 defines product and features, Section 2.3 welding technology, Section 2.4 knowledge based engineering, Section 2.5 enterprise modelling, Section 2.6 enterprise decision making, Section 2.7 new product introduction and Section 2.8 summary of the chapter.

2.1 Engineering Cost Estimation

Cost estimation is an important activity in today's manufacturing industry considering the competition that exists amongst companies. However, the existence of cost modelling literature, particularly in the area of estimating the manufacturing cost of a design is very limited (Langmaak et al., 2013). The manufacturing cost of a design which is also known as unit cost is achieved using parametric process time estimation in combination with a bottom-up calculation of the resources consumed by all manufacturing activities within a process. Langmaak (Langmaak et al., 2013) concludes that the £-per-hour cost rate of every manufacturing operation is realized by the bottom-up costing element and the parametric element of the model is also realized by linking historical operation times and design data for an operational time estimate for future products based on its design parameters. The prediction of the unit cost of a future design is estimated by multiplying the regresses operation times by the respective cost rates and summing up the results of the costs. (Westkämper et al., 2005) asserts that in this information technology age, there are a plethora of software available in the market that is able to help with cost in the manufacturing environment, but each comes with different purchasing costs, benefits, and implication of possibilities and risks. A research conducted by Volkmann (Volkmann and Westkämper, 2013) on cost model for digital engineering tools for the analysis of the variants of process costing methods, however, concludes that accounting methods do not focus on the calculation of cost before the introduction of a new product but rather usually based on data retrieved after implementation. According to Curran (Curran et al., 2004), cost estimating could be defined as the means of predicting the cost of a work activity by making sense of historical data or knowledge by means of creating a cost model. Cost estimation is an industrial practice which is largely based on the experience of the cost estimator and not on science due to its lack of consolidating theories. However, other unpredictable factors outside of design, such as inflation and market condition have the tendency of affecting the costing process (Curran et al., 2004), (Scanlan et al., 2002). However, in some industries such as the aerospace industry which is a high-tech but low-volume manufacturing, obtaining well documented and comprehensible costing information becomes very challenging (Curran et al., 2004). Objective cost estimation and validation becomes a major challenge in situations where there are meagre and increases in inaccurate data (Collopy and Curran, 2005) (Smith and Mason, 1997).

2.1.1 Cost Estimation Techniques

Estimating cost and cost modelling helps to know the future manufacturing cost at very early stages of a product design. Estimating the cost during early design stage with enough accuracy has great benefits to a project launch (Hueber, 2016). Having a poor cost estimate may result in too low or too high cost values which can eventually cause financial loss or loss of order to an organization.

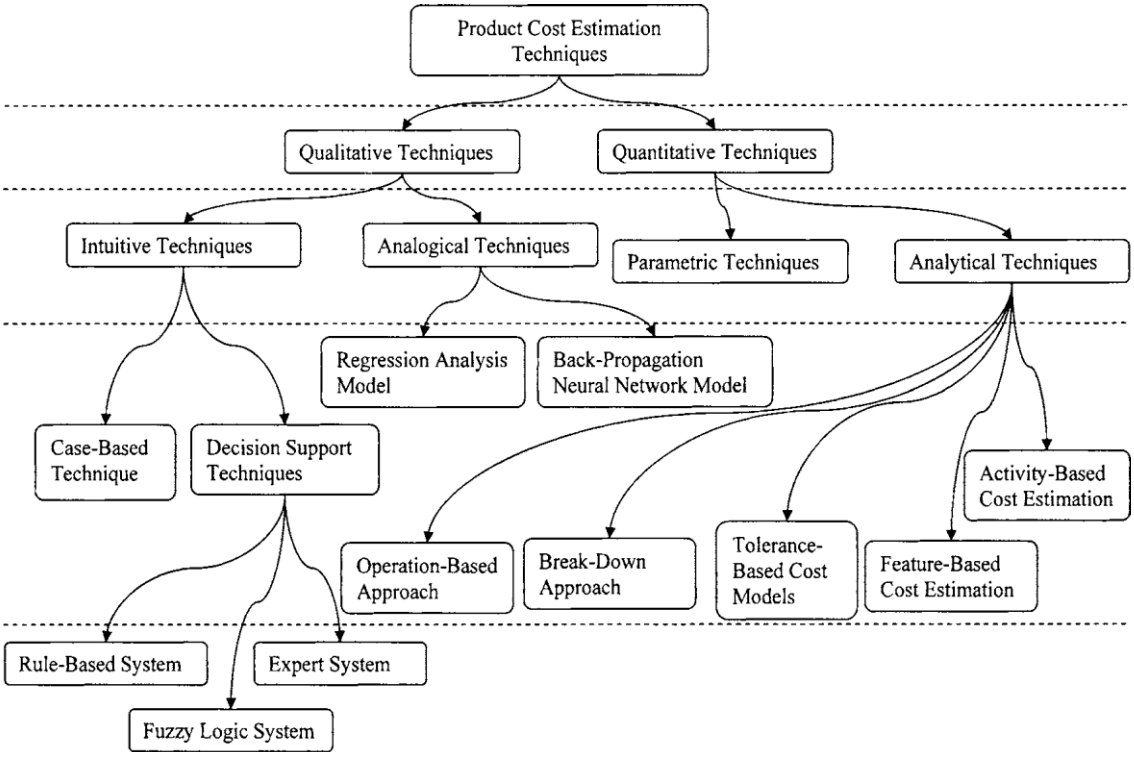


Figure 2. 1 Product cost estimation techniques (Niazi, Dai et al. 2006)

Figure 2.1 shows the various classifications of product cost estimation techniques used in the field of engineering. In view of this research, qualitative costing techniques are most suitable considering the objectives of the research where the cost of engineering changes are to be captured and translated numerically. Under the quantitative group of techniques, the most appropriate techniques to be reviewed and analysed are the parametric, activity-based and feature-based cost estimation techniques.

2.1.2 Activity-Based Costing

The Activity-Based Costing (ABC) methodology was first introduced by (Cooper and Kaplan, 1987) as a more efficient way of costing compared with the traditional accounting methods. A comparison of ABC and the known traditional cost accounting methods was done by (Park and Kim, 1995). Their research claims that the main advantage of ABC is its ability to more accurately reflect indirect costs of different products. However, (Park and Kim, 1995) concluded that it takes more effort in obtaining accurate information concerning the consumption of resources at each activity of a process and their cost driver rates. Lewis (Lewis, 1993) came up with a definition for ABC as: “a method for accumulating product cost by determining all cost associated with the activities required to produce the output”. More critical, (Lewis, 1993) spotted two differences between ABC system and the traditional system; firstly, cost pools are defined as activities rather than production cost centres and secondly, the cost drivers used to assign activity costs are structurally different from those used in traditional cost systems. ABC gained popularity over the years due to dissatisfaction with the distortions created by traditional costing systems and has a wide usage both with accounting academics and in business practice (Cooper and Kaplan, 1987). According to (Turney, 1991), ABC model consists of a ‘*cost assignment view’* and a ‘*process view’* which has ‘*activities’* in between the two views. In the cost assignment view, information on resources, activities, and cost objects are provided whiles the process view gives both non-financial and financial information regarding cost drivers and key performance measures and indicators for each activity or process. The consumption of resources by products based on ABC could be analysed by modelling the causal relationship between products and resources used during a production process. ABC then gives a good understanding of process activities that demands costs as well as providing more accurate cost information on products to help in decision making (Cooper, 1988). Figure 2.2 shows the components of the ABC model, showing the relationship between cost objects, resources, resource drivers, activity drivers, activity elements, activity cost drivers and activity centre.

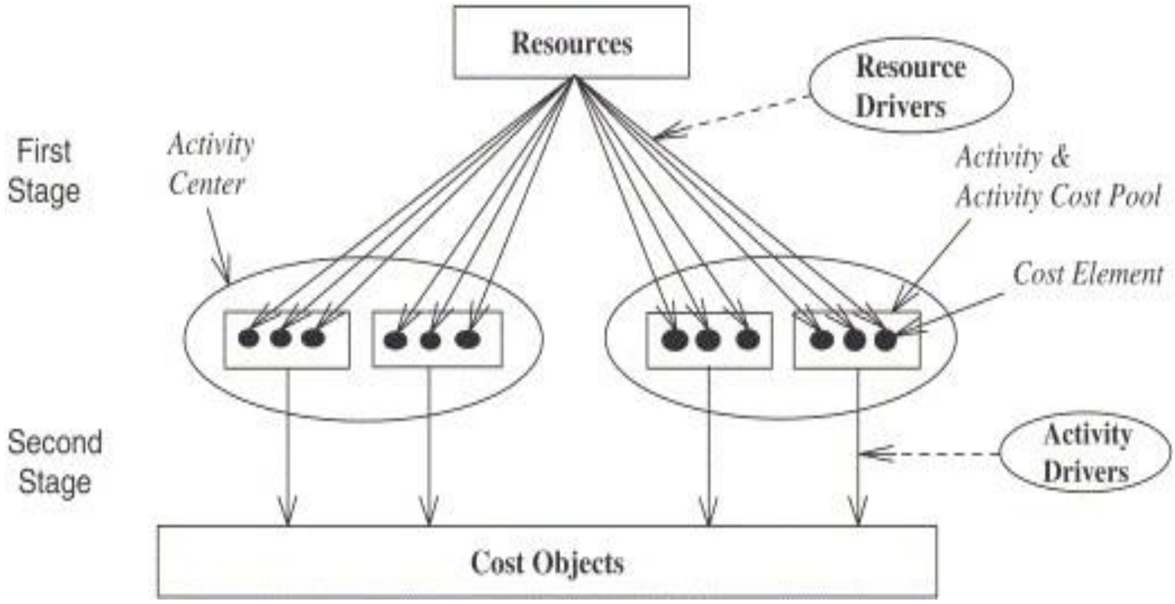


Figure 2. 2: Activity Based Costing Model (Turney, 1996)

2.1.2.1 Activity Based Costing in Manufacturing

ABC analysis model was developed and used by (Tseng and Jiang, 2000) to help in evaluating the different manufacturing costs for multiple feature-based machining methods, restricting the shape of the part to prismatic parts and prismatic features suitable for machining on 3-axis milling centre. Tornberg (Tornberg et al., 2002) also combined ABC and process modelling for the evaluation of various design options in cost. In their research, they modelled processes using graphical flowcharts form design, purchase, to manufacturing. For design and development activities for machining rotational parts, (David Ben-Arieh, 2003) adopted activity based costing technique for cost evaluation. ABC methodology was adopted and used in push and pull manufacturing systems for the estimation of manufacturing and production cost where the two systems were compared to verify their effects on cost (Özbayrak et al., 2004). A product cost-estimation framework was proposed by (Park and Simpson, 2005) in support of product family design based on ABC consisting of three phases: allocation, estimation and analysis. Other research also suggests the combination of ABC with two or more approaches for quicker and more accurate cost estimation (Niazi et al., 2006), (Tornberg et al., 2002). According to (Gunasekaran and Sarhadi, 1998), new concepts in manufacturing technologies including design for quality, design for production and design for distribution which aims at eliminating the non-value-adding activities in processes has their roots from ABC methodology. The implementation of ABC method of costing is confirmed in five different firms in Finland ranging from metal and engineering to food industries. The research looked into the following: (i) the reasons why the companies wanted to implement ABC, (ii) the cost drivers for different manufacturing environments, (iii) the implementation issues including behavioural and managerial aspects, and (iv) the results obtained from the implementation of ABC (Gunasekaran and Sarhadi, 1998).

A research conducted by (David Ben-Arieh, 2003) adopted and modified (Cooper and Kaplan, 1999) ABC model, stating the implementation process of the activity based costing methodology. According to them, the implementation follows the following steps:

i) Identify cost centres
ii) Analyse indirect costs and calculate their cost-drivers rates
iii) Assign resources to each cost centre and determine cost centre driver rates
iv) Identify activities
v) Analyse each activity and find the total cost for each activity
vi) Define activity drivers for each activity and find activity cost-driver rate
vii) Estimate the cost of new parts via activity cost-drivers spent

Limitations of ABC - Limitations of the ABC technique noted by (Roy, 2003) are: it is time-consuming when properly implemented and it is expensive to implement and operate in an entire company. Other disadvantages identified using ABC its difficulty in making it the only costing method. Allocation of overhead cost is complicated and requires experienced personnel. Jawahar (2009) also mentioned that due to the numerous cost pools and multiple cost drivers of ABC systems, its use of may be more complicated when compared with the traditional product costing systems. Due to the complexities involved in the use of ABC systems, Saxena, et al, (2010) is of the view that more time is needed for the analysis of the activities taking place in the activity centres, tracing of cost to activities and determining of cost drivers.

2.1.3 Parametric Cost- Modelling Technique

The parametric cost-modelling technique is often associated with (Wright, 1936) in some of his initial works which he was able to determine the cost of an aircraft as a function of the total quantity of aircraft manufactured. According to Crawford (Crawford, 1985), wright's modelling technique was extended by Stanford Research Institute and then further extended by Rand Corporation. The advancement of parametric cost modelling accredited to Rand Corporation according to Crawford (Crawford, 1985)and (Curran et al., 2004) are as follows:

i) The developing cost estimating relationships (CER).
ii) Merging the CER and the learning curve to form the base of parametric estimating.
iii) The development of CERs for aircraft cost as a function of speed, range and altitude.
iv) The observation of statistical correlation in checking CERs.
v) The development of families of curves data for different aircraft.

When developing parametric cost models, Pace (1995) suggests three most important relationship factors that have to be considered which are:

i) Performance and physical attributes are measures of technical capabilities.
ii) Technical risk and design maturity attributes of a part are measures of the relative difficulty of developing and producing the part.
iii) Programmable parameters describe ways in which programs are operated (Agyapong-Kodua and Weston, 2011).

Parametric cost- modelling technique is being used in different engineering discipline such as in the construction (Akintoye and Fitzgerald, 2000), aeronautics (Polmear et al., 1999) and component part manufacturing (Curran et al., 2004) industries for their design, development and production or implementation phases of engineering projects. (Agyapong-Kodua et al., 2011) however, concludes that the cost model is useful for bidding and target cost estimation as well as for determining the cost of components manufactured.

The principle behind the parametric method is the collection of collection of historical cost data and converting it to mathematical forms known as cost estimating relationship that may be used for estimating similar activities in future projects. As mentioned earlier, the cost estimating relationship CER is the backbone of parametric cost model for estimating resource needs in projects. According to the International Society of Parametric Analysts (ISPA), CERs expressed mathematically are mostly in the form of algebraic equations although sometimes they may be tabulated data (ISPA, 2008). Figure 2.3 below shows a generic process flow of parametric cost modelling technique.

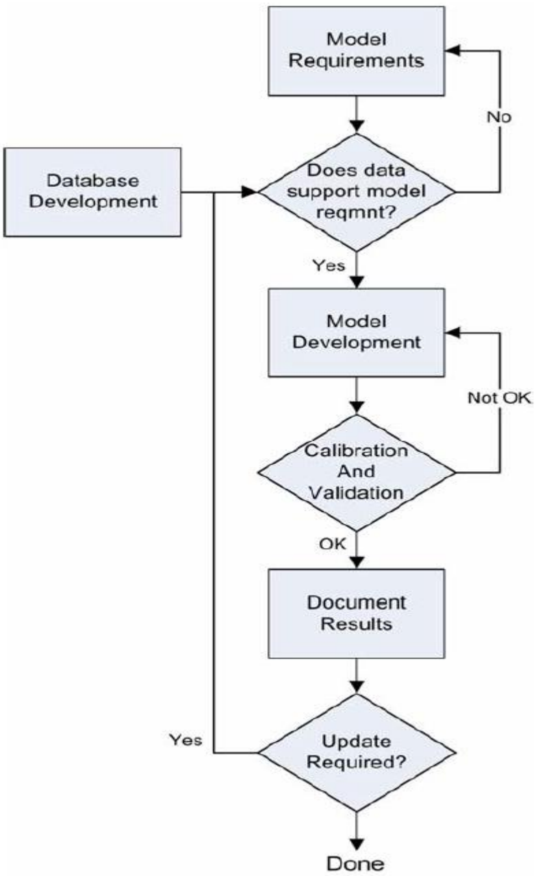


Figure 2. 3: Parametric Cost Modelling Process (ISPA, 2008)

Suggested steps involved in the development of parametric cost models adopted from (ISPA, 2008) are as follows:

i) Develop a database;

ii) Identification of Model Requirements;

iii) Resolution of model architecture and data availability;

iv) Development of model based on requirements and defined assumptions;

v) Calibrate and validate the model for its credibility;

vi) Model documentation;

vii) Model updating.

Limitations - A major limitation of parametric cost methodology identified is the availability of adequate quality data (Beltramo, 1988), (Briand et al., 1999), (Shepperd and Cartwright, 2001). More critically, (Gray et al., 1999) concludes that expert judgement is necessary because, in real life situations, poor and inadequate data is always the case. Also, (Beltramo, 1988) argues that the cost modelling technique is incapable of predicting the cost of projects or products beyond the scope of CER databases. Roy (Roy, 2003) also identified a couple of limitations associated with parametric costing techniques such as parameters not included in costing can become important, mainly useful when combined with other methods, CERs are sometimes too simplistic to predict costs accurately and also, uncertainties are high as CER specifications are not available. In situations where cost drivers cannot be identified, the technique is ineffective (Cavalieri et al., 2004), (Hajare, 1998), (Roberts and Hermosillo, 2000), (Boothroyd and Reynolds, 1989).

2.1.4 Feature Based Cost-Modelling Technique

A feature is defined by (Shehab and Abdalla, 2001) as "a generic shape carrying product information, which may aid design, or communication between manufacturing and design, or between other engineering tasks such as assembly, manufacturing, and maintenance". Features are classed into two categories; design related such as the material type used for a particular product, geometric details, etc. or process oriented, where a product to be manufactured is dependent on a particular manufacturing process such as machining, casting, injection moulding etc. (Niazi et al., 2006) (Agyapong-Kodua et al., 2011) described feature based cost-modelling technique as a product design approach which links product design features to the cost of the product. Their description of the costing technique is based on the fact that, (1) functions that describe cost can be derived from classes of similar objects and (2) factors which affect cost can be influenced by design decisions (Wierda, 1991). Apparently, this costing technique is used in many research investigating the integration of design, process planning and manufacturing ((Bronsvoort and Jansen, 1993) (Ou-Yang and Lin, 1997)). Feature-based cost estimation methodology is also described by (Niazi et al., 2006) as the identification of a product's cost-related features linking it to its associated costs. Rush (Rush and Roy, 2000) concluded that the increase in the use of CAD/CAM technology and 3D modelling has possibly influenced the development of feature-based costing technique. (Curran et al., 2004) also argues that majority of manufacturers have a database of historical geometric data related to product features which could be linked to technical specification through functionality and performance, and manufacturing capability.

Limitations - Curran (Curran et al., 2004) has observed several limitations of the technique. According to them, although feature based cost modelling technique is understood by many many design and manufacturing engineers, the concept has still not gain industry wide application. Another challenge they identified with this technique is that when product features changes, it becomes very tedious to redesign a cost model that captures the changes. That is, the cost of engineering changes on a product in the form of a feature modification is not automatically captured by this technique. Another limitation of this technique it is not capable of estimating the cost of complex or very small geometric features on a product where machining processes are required to produce these features (Niazi et al., 2006).

2.1.5 Cost Modelling: Accounting versus Engineering

Literature shows that estimation of product, process and resource cost has been mainly achieved through traditional cost accounting methods usually led by Accountants, Business Administrators and Economists (E. Shehab & Abdalla, 2002; Son, 1991). Despite their long industrial adoption, traditional cost accounting models are usually intended for management and financial appraisal and do not directly reflect the cost implication of engineering decisions (Agyapong-Kodua, Asare, & Ceglarek, 2014; Johnson & Kaplan, 1987; Maskell, 1991). Consequently, current generation independent cost accounting models perform less well when applied to dynamic product and process design scenarios (Agyapong-Kodua & Weston, 2010). A typical scenario is in the space industry on continual cost overruns of engineering projects, an issue which has existed for the last 40 years although there are better databases, models, estimators, and more stringent reviews (Arthur et al., 2004). According to audit experts on such industry, such as US General Accountability Office, who track cost performance there is difference between models for predicting cost at the inception of projects and those for project implementation (ref). This raised arguments where some are in support of the accuracy of the costing tools and models yet accepts the fact that essential cost elements are omitted whereas others disagree and argue that the tools used are inadequate and a better approach to costing is needed (Keller, et al., 2014). More critically, at early stages of engineering product design, designers need to understand the cost implication of their decisions. Engineering decisions based on CAD models can relate to tolerances, materials selection, dimensions, speed, assembly sequence, cycle time, etc. Traditional cost accounting does not provide formalisms for such micro level cost estimation and it currently rests on the experience of the engineer. The situation becomes more challenging, when there are different technical options to choose from. Under such situations, there must be a fair balance of technical and economic indices and a confirmation of which outcomes to trade-off. This is where cost engineering knowledge becomes very useful. Also in establishing budgets for technical projects, preparation and evaluation of price proposals, contract negotiations and assessing the cost impact of introducing engineering changes to existing designs. To achieve these, cost engineering techniques attempt to capture practical experience, analyse the experience in order to develop tools and models which, together with expert judgement, can be applied under different circumstances to make predictions of likely cost or assessments of whether a proposed cost is reasonable (David, Herve, & Cahill, 2003). Many authors (Johnson & Kaplan, 1987; E. Shehab & Abdalla, 2002; Son, 1991) have argued that cost engineering mainly focusses on cost estimation and control but latest research activities from the cost engineering domain has shown that cost engineering extends beyond estimation and assessment of cost but includes engineering knowledge which in general terms can help achieve cost effective solutions. Nonetheless, there is the need for scientific methods and techniques to support cost engineering activities so that timely and relevant results can be generated at all times (Agyapong-Kodua, Wahid, & Weston, 2011; Rush & Roy, 2001b). In view of this some researchers in enterprise and systems engineering domains (Kwabena Agyapong-Kodua, J.O Ajaefobi, & R H Weston, 2009; Baines, Harrison, Kay, & Hamblin, 1998; Bernus & Nemes, 1996; Kosanke, 1996; Weston, 1999) have indicated that there is the need for well-structured process-based models to support in-depth analysis.

In recent times, cost engineering principles and modelling methods have been applied by some researchers to support cost estimation, business analysis and planning, project management, profitability analysis and scheduling of major engineering projects (K. Agyapong-Kodua, J. O. Ajaefobi, R. H. Weston, & S. Ratchev, 2012; Agyapong-Kodua, Asare, et al., 2014; R Curran, S Raghunathan, & M Price, 2004; Tammineni, Rao, Scanlan, Reed, & Keane, 2009). Many researchers (Agyapong-Kodua, 2009a; Agyapong-Kodua, Asare, et al., 2014; Agyapong-Kodua & Weston, 2010; Curran, Watson, Cowan, Mahwinney, & Raghunathan, 2003; Roy & Palacio, 2000; Rush & Roy, 2000a) have indicated that to overcome the limitation of cost estimation imposed by traditional cost accounting techniques as well as meeting time deadlines, there might be the need for (1) first sight estimate (suitable for 'Rough Order Magnitude' and (2) detailed estimate for precision costing.

More critically, in general terms, companies are expected to meet customers' requests for quotation in a more efficient and faster manner. The ability to satisfy such requirements is an enabler for good competition and usually a determinant for a company to survive economically. Meeting the requirement for product or project quotation is usually faced with the problem of over or underestimation since in most cases, actual manufacturing systems behaviour or capacity to meet customers' requirements cannot be fully estimated at an early stage of product development. This is even more critical for new product development but the ability to satisfactorily predict cost within reasonable limits of accuracy partly determines an industries (particularly for engineer-to-order enterprises) ability to maintain the lead in product development. In view of cost estimations, underestimation may result in an industry losing money whilst overestimation may result in loss in competition (Veeramani and Joshi, 1996). For these reasons, industries strongly desire a fairly accurate cost estimation solution in support of design activities. Wierda, (1990) concludes that proper costing models can successfully improve the performance of these two strategic functions of organisations. Also, traditional approach to product design and analysis usually requires a designer to first develop a design solution which is then passed on to manufacturers and process designers to provide input into the manufacturing feasibility. When the product designer and manufacturers have agreed on a common solution then the design is passed onto an estimator to calculate the cost of implementing the solution. This can make the design cycle very long and expensive since a lot of decisions would have been confirmed already before passing on the design to the estimator (Rush & Roy, 2000b).

2.1.6 Limitations of The State Of The Art Techniques

Despite the introduction of cost modelling techniques and IT systems in support of engineering decision, there are yet some identified limitations across various domains. According to Collopy and Curran (2005), there are four major challenges with modelling cost in the aerospace industry which includes the complexity of the cost, the need for cost model validation, presence of cost drivers outside the designs, and non-objectivity of estimates in some cases. Within the oil and gas industry, Hall and Delille (2011) identified escalation of prices a unique factor which makes operations unpredictable to estimate cost with uncertainties. A research in operational processes in-situ of three software development companies conducted by Ramasubbu and Balan (2012) revealed three challenges faced in cost estimation during early design stages. Firstly, diverse characteristics and configurations of different projects were not factored in standard cost estimation tools. Secondly, detailed data are required by cost estimation tools which are not always available at the product design stage. Finally, expert knowledge and involvement is required to generate good cost estimates.

More critical, this research is of the view that there is a lack of integration between domain experts and digital cost estimations tools. Although standard cost estimation tools are capable of estimating known processes, they are not capable of estimating the cost of emerging processes. This is because estimation tools have inbuilt databases of processes, resources, cost equation, material, scenarios, etc., which requires current and adequate data to maintain and update them. Also, updating cost estimation tools requires interaction with the tool's database to develop new rules, logics and options. The maintenance activities are usually done by IT experts where the knowledge of doing this is not clearly documented for engineers to follow. The author is of the view that capturing the knowledge of extending the capabilities of cost estimation tools to include new process options will give engineers much flexibility of customising processes to support engineering decisions.

2.2 Product and Features

There are lots of research work done on computer aided design (CAD) particularly in the area of product feature extraction. Most of these research uses design interfaces such as DXF, IGES or others. In the last few years, research focused on CAD/CAM technology consider Product Model and its associated design interfaces. According to Arunkumar et al. (2009), a Product model contains geometric as well as the technical information embedded into it.

International Graphics Exchange Standard (IGES) has been used as a standard for translating CAD/CAM over the years for moving two-dimensional models from one program to another. IGES is powerful in transmitting basic CAD model geometry, however, the Standard for the Exchange of Product Model Data (STEP) has also attracted attention in recent years. STEP gaining popularity is as a result of it going beyond just transmitting CAD geometry but also providing the ability to express and exchange digitally useful product information during the Product's design, analysis, manufacturing, and even support. STEP aims at creating a single international standard covering all aspects of CAD/CAM data exchange (Owen, 1993; Nazemetz and Ravat, 2003).

CAD and CAM tools are developed by different companies which come with different formats and standards and this limits seamless interactions during operation. According to Jones et al. (2011) CAD software design type features, however, may not be equivalent to manufacturing features of CAM where design is seen as an addictive philosophy and manufacturing is subjective.

2.2.1 Feature Definitions

Cayiroglu (2009) claims that the definition for feature in the manufacturing domain is not straightforward. Garcia et al. (2011) however describe a feature as an element of a mechanical product CAD that has a specific functionality and that is semantically significant within the manufacturing process. Farsi & Arezoo (2009) also classified features on sheet metal into internal and external features. According to them, internal features are holes and external features are different types of notches. Another classification of features was done by Radhakrishnan et al (1996) who claims that features are grouped into two rule types: intrafeature rules and interfeature rules. They further explained that intrafeature rules are related to constraints that indicate minimum values of dimensional parameters of the feature itself whereas interfeature rules elaborate on the restrictions across features and contours. A feature is also defined by Sreevalsan & Shah (1992) as an entity used in reasoning about the design, engineering, or manufacturing of a product. Other researchers defined a feature as a geometric form or entity whose presence or dimensions are required to perform at least one CIM function and whose availability as primitive permits the design process to occur'' (Devireddy & Ghosh, 1999; Luby, Dixon, & Simmons, 1986). Pratt & Wilson (1985) also concludes that a feature is a region of interest on the surface of a part.

2.2.2 Feature Recognition

CAD feature recognition has over the past thirty years attained much research attention. Recognising CAD model features aims at extracting certain substructures from a solid model (Zhu H, Menq C, 2002; Mäntylä et al., 1996; Han et al., 2000). Babic et al., (2008) are of the view that computer-aided process planning (CAPP) applies feature recognition for the generation of sequences of instructions for manufacturing. More critical, CAD feature recognition is a fundamental task of extracting and identifying information contained in a CAD model. Traditionally, CAD feature recognition and extraction were accomplished by manual human techniques which is time consuming for a complex product and also generated errors. In more recent times, automated feature recognition can best be facilitated by CAD systems capable of generating product feature geometries and at the same time capture and store those features.

2.2.3 Feature Recognition Techniques

Since the classification of geometric CAD model by Kyprianou (1980), many research works have considered recognising CAD features. Han et al. (2000) and Babic et al. (2008) further classified features recognition systems into volumetric decomposition, graph-based, hint-based.

Volumetric decomposition – this is a general way and yet a better means of recognising interactive features by decomposing a CAD model into a set of intermediate volumes and manipulating the volumes to produce CAD features (Little et al. (1998) and Han et al. (2000)). For this technique, a convex hull must first be determined around a given part. The difference between the volume of the part and its convex hull is known as the alternating sum of volumes (ASV). This technique was initially applicable to polyhedral parts due to the complexity of convex hull computation of curved parts. However, Kim (1992) overcame this by introducing a remedial partitioning procedure (ASVP).

Graph-based Approach – this approach was first introduced by Joshi (1987) represents the topological and geometric information of an object using graph structure-usually a structure obtained from (or embedded in) the data structure of the boundary representation of the object. Other researchers break this approach into two categories: based on graph search (Corny and Clark (1991)) or based on pattern matching (Joshi and Chang (1988); Pinilla et al (1989); Scott (1990)). The graph-based approach makes use of an attributed adjacency graph (AAG) for transforming B-rep part models.

AAG translates arcs that have concave node adjacency relation into “0” and “1” if a part has a convex adjacency relation as shown in Figure 2.4.

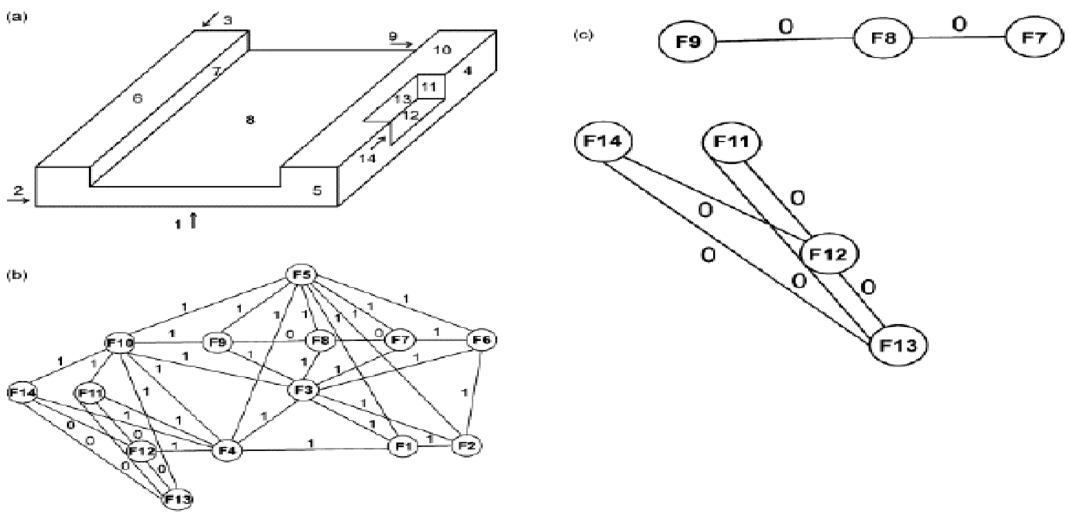


Figure 2.4: (a) Sample part; (b) Attributed graph of the part; (c) its subgraphs ***(***Owodunni et al., 2002)

Hint-based approach – this approach is more efficient for small features yet, it is dependent on how the hint is generated and defined and it refers to hard-coded features which may be challenging for end users modify or define new features (Fu et al. (2003); Shah et al. (2001)). Hint-based was intended to tack interacting features recognition problems, by assuming that certain features patterns such as faces, edges or vertices exist in the solid model of a part even though some other attributes may be modified or removed by the interactions. The existence of feature patterns has been used for the generation of hypotheses about the presence of features in a part model. This approach was initially investigated by Vandenbrande and Requicha (1993) and Marefat and Kashyap (1990) with further development by Ames (1991); Regli (1995); Gao and Shah (1998) and Li et al (2000). Brousseau et al. (2008) identifies two limitations of this technique: (1) majority of its application are restricted to the domain of machining features and (2) the challenge of developing an appropriate set of hints for each considered application domain.

2.2.4 Features Representation

Recognising CAD features is an important aspect of most pattern recognition systems that requires detection (Shah et al. (2013)); recognition {Shah et al. (2016), Zhang (2015), Cui et al. (2015)}; registration (Shah (2010)); reconstruction (Afonso and Sanches (2015)) and classification (Yoon and Friel (2015)) of solid objects. All the engineering domains have their own view on how they define features. For example, designers may represent features as functionalities; a machinist may represent feature as the effect of a cutting operation; an assembly planner might view features as region of a part that has to be connected to a corresponding feature of another part or component; an inspection planner might also view feature as a pattern of measurement points. There are, however, three main solid representations of CAD features in research which are Constructive Solid Geometry (CSG), Boundary Representation (B-Rep) and Spatial Subdivision {Requicha and Rossignac (1992) and Hoffmann and J. R. Rossignac (1996)}.

Boundary Representation (B-Rep) – this represents a solid geometric CAD object as the intersection of halfspaces represented by the surface of the solid. This is commonly used in most commercial CAD systems as an internal kernel due to its flexibility and speed for user editing of object shapes. The concept of B-Rep is based on the topological assumption that every physical object is bounded by a set of faces and the faces are regions or subsets of closed surfaces and surfaces which can be oriented (Shah et al., 2016). B-Rep uniquely defines CAD model entities such as faces, edges and vertices and link them together in a manner that ensures the topological consistency of the model.

Constructive Solid Geometry (CSG) – CSG is formerly called computational binary solid geometry, an approach adopted in solid modelling, making it possible to construct complex solid models through primitives, boolean operators and rigid motions via merging solid primitives or subtracting the primitive of solid models. With CSG, both the interior and surrounding surfaces of an object are fully defined. The uniqueness of this approach is that its building blocks are solid primitives such as blocks, cylinders, spheres, cones and tori, defined in the world system of coordinates.

Once primitive objects are combined successfully, regularised Boolean operations denoted by $\cap^*$ for regularised intersection; U^* for regularised union; and -* for regularised difference takes place (Ullrich, 2016). CSG object may be represented in the form of a tree with the leaves as primitives, nodes as solids, edges as Boolean operations and roots representing a solid CSG object (e Geuzaine et al., 2015) as shown in Figure 2.5.

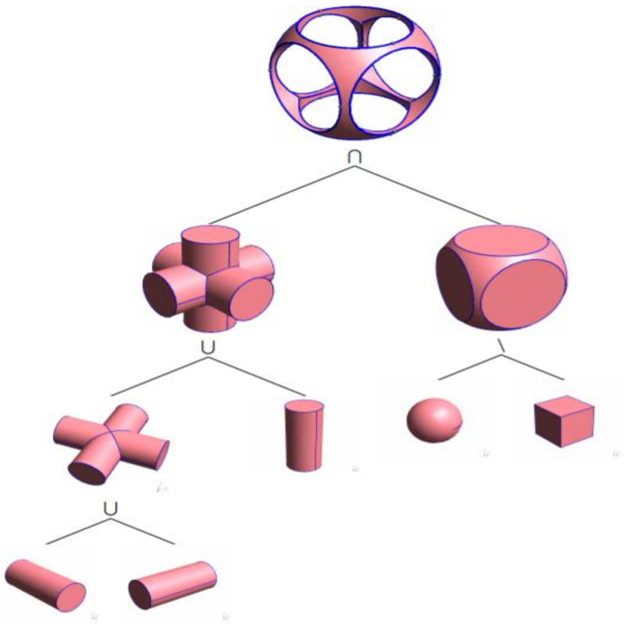


Figure 2.5: An example of CSG Tree *(Geuzaine, 2015)*

Shape achieved optimization using CSG representation (Kodiyalam et al., 1992) whereas Chirehdast and Papalambros (1994) also claim the use of the concept for realizing the complex geometry resulted from topology optimization.

2.3 Welding Technologies

There are various welding techniques used in sheet metal fabrication, particularly within the automotive industry. These welding techniques are carried out manually which may include manual spot welding and MIG welding or with the use of robots (welding and MIG) in industrial settings. This section reviews resistant spot welding (RSW) and remote laser welding (RLW) techniques.

2.3.1 Resistant Spot Welding (RSW)

For several decades now, resistance spot welding (RSW) has gained a wide application in sheet metal fabrication due to its cost-effectiveness and simplicity of its technology (Nied and Zhou, 1984; Hao et al., 1996; Khan et al., 2000 and Santos et al., 2004). According to Wan et al., (2014), RSW was first introduced in 1877 by Elihu Thomson, a technology which is currently used in the automotive industry for body in white welds. Moshayedi and Sattari-Far (2014) describes four steps involved in RSW as squeezing step, electrical current transmission, holding step and finally, workpieces are release and cool down. Davies (2012) also came up with a similar description of RSW process in his research on component assembly: materials joining technology. In the automotive industry today, the use of robots to aid in carrying out RSW tasks is the current state of the art. Robotic end effectors for double sided RSWs are a C-type gun which hinders access to weld spots typically on the auto body shop floor. To overcome this and many other challenges, Poss and Lendway (1997) introduce a new welding system designed to create a weld using single-side access with low electrode force. More critically, Kim et al., (2013) conclude that although it is expected that the use of this method may be practically successful, the basic characteristics of the welding method are not well understood. Despite RSW has being an accepted welding technology in industry over the years with numerous advantages in its flexibility and adaptability for automation, low cost and high welding process speed, there are yet some limitations for this technology, which includes robot reachability and accessibility to the spot to be welded in case a complex fixture design and its high cycle time as well.

2.3.2 Remote Laser Welding (RLW)

Havrilla (2014) of TRUMPF concludes that there is a vast array of application of the laser welding technology which may include the powertrain application (Figure 2.6.) and tube welding (Figure 2.7)

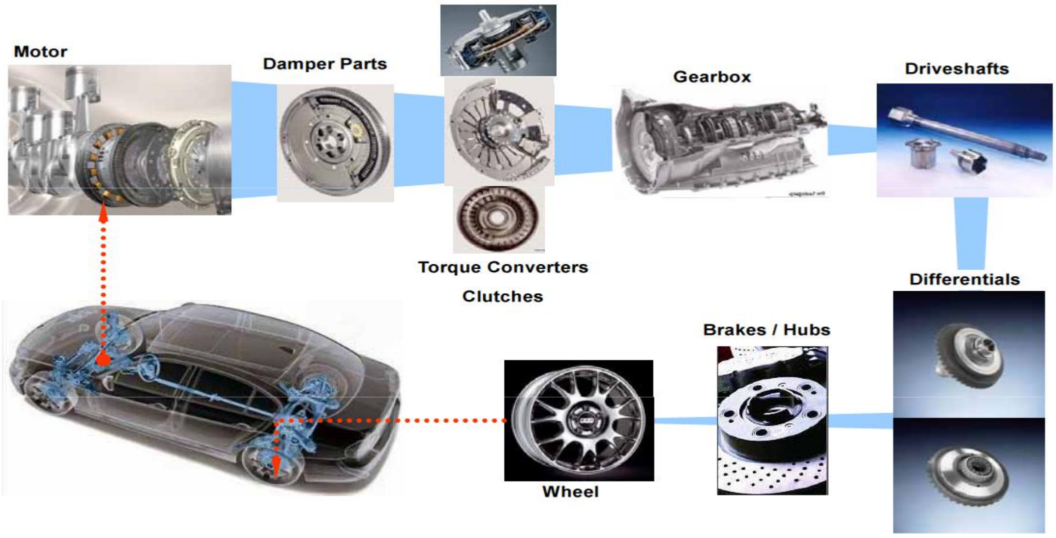


Figure 2.6: Powertrain Application

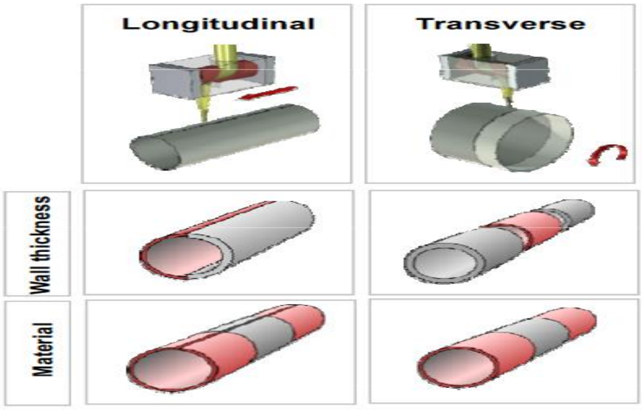


Figure 2.7: Tube Welding

The limitations of RSW gave birth to research into other welding technologies, particularly for remote laser welding (RLW) technology which is a current trend in the automotive fabrication. An initial experiment conducted in the mid-1990s to analyse keyhole welding application which has a focal length of up to 1600mm was described as "Remote laser welding" (Macken, 1996). Hence, Zaeh et al., (2010) describe the remote laser welding technology as a laser beam deep penetration welding process which applies laser beam from a distance. Generally, the RLW process consists of a laser head which transfers heat through a laser beam ranging from 1m to 1.6m and a scanner system containing two mirrors attached to the end effector of an industrial robot to deflect and directs the laser beam (Zäh et al., 2006; Tsoukantas et al., 2007). The reflected laser beam is then directed onto the area to be welded. Significant benefits can be gained through the unique characteristics of RLW process when compared with conventional RSW process, which includes more flexible weld design as only one side access to welding area is required, elimination of direct tool to workpiece contact, better weld quality obtained in terms of weight (reduction of flange sizes), high stiffness and low thermal distortion (Davies, 2012; Gabor et al., 2012).

This technology has moved from research into industrial production application since the past two decades (Davies, 2012). The application of RLW technology was exhibited at Bad Nauheim 2004 at the Laser Application Conference where typical automotive applications were cited by manufacturers such as VW Golf V who claims 70000mm laser welding and brazing, and Audi A3 demonstrated their extensive underbody welding using laser technology with associated quality control systems. To add to that, BMW also used laser-seam welding extensively on its 6 series mainly on its roof/side frame joint to ensure contact of mating surfaces (Davies, 2012).

According to Quintino et al. (2012), to generate a keyhole on a workpiece in an RLW process, the intensity of the focused laser beam on the workpiece must exceed the threshold value of 10^6 W/cm2 for iron-based materials. For aluminium and its alloys, the threshold value of power density is about 1.5 X 10^6 (W/cm2) due to their higher thermal conductivity and high coefficient of laser radiation and reflection. Due to the high risk of this intensive laser beam, there has to be a dedicated enclosed cell for the purpose of RLW activities. This cell must contain all apparatus needed for the RLW process such as an industrial robot with the end effector, a laser source, a cooling device, fixture device and the workpiece to be welded. The robot control for the welding activity is done outside the dedicated cell. In this research, the cell is designed such that welding operation cannot be carried out when there is an operator in the cell or when the cell door is not properly shut. The enclosed and confined workspace of RLW process brings challenges for robot programming, accessibility and collisions checking. Various studies have been conducted to deal with these difficulties, such as the RLW workstation configurator is looking at the cell configuration design, robot path planning and optimization, accessibility analysis and collision checking (Gabor et al., 2012).

Meanwhile, the quality of RLW process is affected by various parameters: laser power, focus diameter, focus length, working distance and laser beam moving speed, which has an overall effect on the weld/stitch quality (Schlueter, 2007). Different laser sources from Nd: YAG and CO2 to Yb: YAG had been compared in terms of power, efficiency, beam quality and feasibility for fibre transmission (Ahmed, 2005). Currently, intensive research has been conducting in RLW process optimization (laser beam parameters selection, fixture layout optimization), RLW process control (in-line monitoring and control) and RLW Eco-Efficiency evaluation (Pasquale/Darek papers) through the RLW Navigator project, which will provide a toolkit to facilitate the process planning, design, implementation and optimization in the application of RLW technology for Body-In-White Sheet metal joining (RLW project ref). There are also some other challenges for the RLW process in a production environment, such as tight gap requirements (maximum 0.3mm in this case) between two joining parts as well as welding strength verification.

There are other applications of laser welding technology identified by experts across other industries which also includes shipbuilding, construction, pipeline supplies (such as oil-country tubular goods), heavy equipment/off-highway, and railroad equipment (www.ManufacturingEngineeringMedia.com).

Recent legislation on the automobile industry on security, lightness and environmental protection is putting much more pressure on them to adopt new material and new forming mode which can significantly reduce weight but maintain functionality and shock resistance (Tian et al, 2005; Wu et al, 2006). According to Qiang et al., (2008), the use of high-strength galvanized steel and laser welding technology becomes the most appropriate approach to achieving this. Huge experiments have being carried out on high speed steel using laser welding technology in the last decade (Park et al., 2002; Tang et al., 2000; Cui et al., 2004; Tian et al, 2005; Wu et al, 2006; Qiang et al., 2008; Holger et al., 2011; Mackwood and Crafer, 2005).

In an experiment conducted by Qiang et al (2008) on laser welding of the vehicle body, galvanised high speed steel (DOGAL 800DP) was used with 1.5mm thickness manufactured in Sweden and general steel sheet BUSD, from Baoshan Iron & Steel Co. Ltd. China. For the purpose of the experiment, the thicknesses of the BUSD material was varied (1.0, 1.5 and 2.0 mm). The gap controlled between the sheet metals to be welded was controlled within 0.10-0.15mm to avoid oxidation of workpiece surfaces during the welding process and to wipe off plasma cloud that absorbs and scatters laser energy during high-power deep-penetration laser welding. Another research conducted by Lifang et al., (2009) on laser welding of high-strength galvanized automobile steel sheets using sheet metal with 1.5mm thickness. According to them, the gap between the workpieces to be welded is critical especially when welding galvanized steels considering the difference between the evaporating temperature of zinc which is 906 degrees Celsius as well as the 1583 degree Celsius for the melting temperature of steel. More critical, Qiang (2006) concludes that the gap between the workpieces should be controlled within 10% thickness of workpiece when deep laser penetration welding is required. Havrilla (2012) agrees to this and adds that 10% gap has to be related to the thickness of the upper part upon which the laser beam is an incident. The gap has to be factored in when preparing for laser welding to avoid problems such as porosity and uncontinuity during welding.

2.3.3 Gap Controlling Techniques

There are about ten means of ensuring that gap between workpieces to be laser weld is maintained as identified as identified by Sinha et al. (2013) which may include ‘shim, Pre-stamped Projection, Laser dimpling technique, etc. more information on current techniques for gap control for laser welding including the benefits and disadvantages of each technique is given in Table 2.1 below.

Table 2.3: Part-to-part gap control techniques in laser welding

Industrial solutions	Description	Advantages	Disadvantages
Shim insertion	Insert shims in between sheets	Intuitive and easy to control the required clearance	Needs additional work and tool for shim insertion
Pre-stamped projection	A preprocessing creates V-shaped tabs in the lower part which act as a gas venting channels that allow zinc vapor to escape during laser welding process	Useful for hem joints (special case of lap joint configuration)	Needs preprocessing
Laser dimpling technique	In this technique a preprocessing is carried out in which the laser beam generates dimples to maintain a part-to-part gap.	Dimples can be produced with the same laser system used for the laser welding	Needs two-step process: (i) pre-processing by which dimples will generate and (ii) actual welding
Fill the part-to-part gap with a porous powder metal	The porous powder metal allows zinc vapor to escape without disturbing molten metal	No need to remove the porous powder metal after welding	Difficult to implement in a real production environment
Pre-drilling vent hole techniques	Pre-drilling vent holes along the welding line allow zinc vapour to escape without causing expulsion of the molten metal	No need to provide part-to-part gap	Time consuming and expensive process
Prior zinc removal techniques	Remove zinc coating from welding zone and then coat the treated zone with nickel in order to provide corrosion protection	Nickel coating not only provides good corrosion resistance but also removes the problem of zinc vapour because nickel has a higher vaporization temperature as compare to the fusion temperature of steel	Preprocessing is necessary
Low power pulsed laser welding	By careful control of pulse energy, pulse duration, peak power density, mean power, and welding speed, zinc gas can be reduced in the pulse mode and effectively exhausted through stabilized keyholes	Literature survey reveals that both CO_2 and Nd: YAG pulsed laser provide porosity/spatter free welds	The limitation of low laser power causes slow welding speed of less than 2.4 mm/s, which is hard to maintain in an industrial environment
Altered joint geometry techniques	Altered joint geometry offers channels between the metal parts to exhaust zinc vapour	This technique is very useful when the dimensional variation in between part-to-part gap is high	Intentionally, we need to create altered joint geometry in the form of either concave or convex on the top surface of the metal part
Dual beam hybrid technique	The first beam creates a slot as an effect of preheating and the second beam performs actual welding process	The first beam facilitates vaporization of the zinc that will prevent weld defects	Needs additional complex equipment
Addition of oxygen as a shielding gas to argon	Addition of a small amount of oxygen (2–5%) as a shielding gas to argon facilitates the zinc to react with oxygen and reduce the effect of vaporized zinc	No need to provide part-to-part gap	The flow rate of shielding gas must be optimized otherwise plasma will dissipates and appears as oxides porosity on the surface of the weldment
Vertical positioning of metal parts	Metal parts are positioned and moved vertically during welding while the laser beam is static and applied to parts horizontally	Vertical position of metal parts allow zinc vapour to escape through the weld zone	Due to difficult positioning and movement of parts this idea has been rarely applied in industry
Synchronous rolling techniques	The additional roller generates pressure between part-to-part gap as well as creates favorable conditions for rapid heat transfer from the upper part	Decrease the formation of brittle intermetallic compound like zinc oxide	Needs additional roller during welding

(Source: Sinha et al., 2013)

A mathematical relationship between sheet metal thickness and the gap was proposed by Akhter et al., (1991). According to them, if the value of the gap per metal thickness is approximately in between 0.2 and 0.3mm, then a quality weld may be expected on can expect on galvanized steel

2.3.4 Stitch Parameters

Zhang (2008) mentioned that the most important factors to help identify the quality of a weld is determined by the width of weld seam and its depth of penetration. Their research shows the width of the bottom weld seams but in practice, it is not easy to inspect the width of the bottom seam hence, they estimated it based on the shape of the top weld seam.

Experiment conducted by Sinha (2013) to investigate the shape complexity of the top weld seam and also the welding quality, for the case of laser lap welding of galvanized steel revealed that if the heat input per unit length is not enough as a result of low laser power, large part-to-part gap or high welding speed, weld seam is not formed sufficiently from the initial melting zone which occurs close to the mating plane of the two metal parts to be joined, outwardly onto the top surface of the upper part. The three welding parameters that have to be controlled in order to a good weld are identified in many research as: Laser power, welding speed and part-to-part gap (gap between two pieces to be welded) (Chen et al., 2011; Rizzi et al., 2011; Tian et al., 2005; Furusako et al., 2003).

2.4 Knowledge Based Engineering

Over the years, companies are faced with competitions for customers regardless of high and sophisticated expectation of customers. This is continually challenging manufacturing enterprises to reduce production cycle time in order to launch a product to customers and new market without compromising product quality. Achieving this may require refining and improving the product development process of a manufacturing system.

During the late 1970s, changes in engineering design techniques from table drawing on papers to computers in a two dimensional (2D) computer environment was a great means of modelling processes much easier. However, about a decade later CAE/CAD/CAM system was introduced, the first solid was modelling system which gave a new dimension to product design and the paradigm of virtual prototyping. Andreasen (1987) records integrating product development process players became possible and improvement opportunities are seen in different forms of which Knowledge Based Engineering (KBE) is one of them. Howard (1998) predicted that KBE will have the same impact for companies in 2010 just like CAE/CAD/CAM system had in the 1990s. More critical, Sandberg (2003) argues that the concept goes back to the 1950s where researchers had the objective of developing systems which has its own intelligence known as Artificial Intelligence (AI). The principle of AI was to develop adaptive strategies for solving a broad spectrum of tasks. This system was a failure because humans could solve simple problems much faster than AI systems.

2.4.1 KBE Definitions

As a concept, KBE has various definitions but Ammar-Khodja et al. (2008) conclude that there is no clear-cut definition of the concept in practice and also confirms that most definitions are similar. A definition of KBE by Baxter et al. (2008) is that it is generally regarded as an umbrella terminology which describes the application of knowledge for assisting or automating engineering activities. Cooper and La Rocca (2007) and Van der Laan (2008) are of the view that Knowledge based engineering makes use of dedicated software language tools (i.e. KBE systems) for capturing and re-use of product and process engineering knowledge in a convenient and maintainable manner. By way of automating repetitive, non-creative design activities and supporting integration during early stage design phase of a process, the objective of KBE to reduce cycle time and cost of product development is achieved. More critical, Bermell-García & Fan (2002) mentioned that the KBE concept is a special type of knowledge based system which mainly focuses on product engineering design activities including analysis, cost estimation, manufacturing, production planning, and sales. They claim that the output of the technology provides a high level of automation and design integration in well defined and complex design environment. Over a decade ago, Chapman & Pinfold (2001) said that KBE is a kind of evolutionary approach in Computer-Aided Engineering (CAE) and at the same time an engineering method that integrates Object-Oriented Programming (OOP), artificial intelligence (AI) and Computer-Aided Design (CAD) technologies which generate benefits to customized or variant design automation solutions. KBE is also defined earlier by Sainter et al. (2000) also maintain the fact that ‘a KBE system can be regarded as a type of knowledge based system that performs tasks related to engineering. KBE systems do not express designs with specific data instances, as ordinary CAD systems do, but with sets of rules that enable the design to apply to large classes of similar parts'. However, their definition only looks at the generative aspect of KBE system and not taking other aspects into consideration. Reddy et al. (2015) introduce KBE as the application of Knowledge Based System to the domain of manufacturing design and production. They also stress that KBE is a system that has the capability of accumulating existing information in the growth of the product and making it available for reuse.

2.4.2 Knowledge Based Engineering Methodologies

A KBE system consists of a lifecycle requiring the identification, capture, structure, formalise and implementing the knowledge. Developing a KBE application requires the use of different tools at the various stages of the KBE lifecycle. According to Reddy et al., (2015) most KBE platforms only support the implementation stage of the KBE lifecycle but not the development process. Developing a good KBE system must be robust in terms of managing, safeguarding and updating the system in a structured manner. The following sections will look at some accepted KBE methodology.

2.4.2.1 MOKA

Methodology and tools Oriented to knowledge based Applications (MOKA) is a well-known and applied methodology which satisfies to a great extent most of the objectives of KBE system. According to Curran et al. (2010) and Klein et al. (2010), MOKA was a European research project set up as an international standard for Knowledge Based System development. MOKA is adopted in the aerospace and automobile industry which serves as a bridge between raw knowledge and KBE platform. MOKA does this by breaking down and stacking up knowledge and associating them with a predefined network of a problem domain for various users to relate with.

The objectives of MOKA project were to;

i) Reduce KBE application development times and cost
ii) Provide a consistent way of developing and maintaining KBE applications
iii) Develop a methodology that will form the basis of an international standard

iv) Provide a tool supporting the methodology (MOKA Group, 2000).

MOKA methodology consists of six steps: Identify, Justify, Capture, Formalize, Package and Activate as shown in Figure 2.8 mainly focuses on capturing and formalizing knowledge.

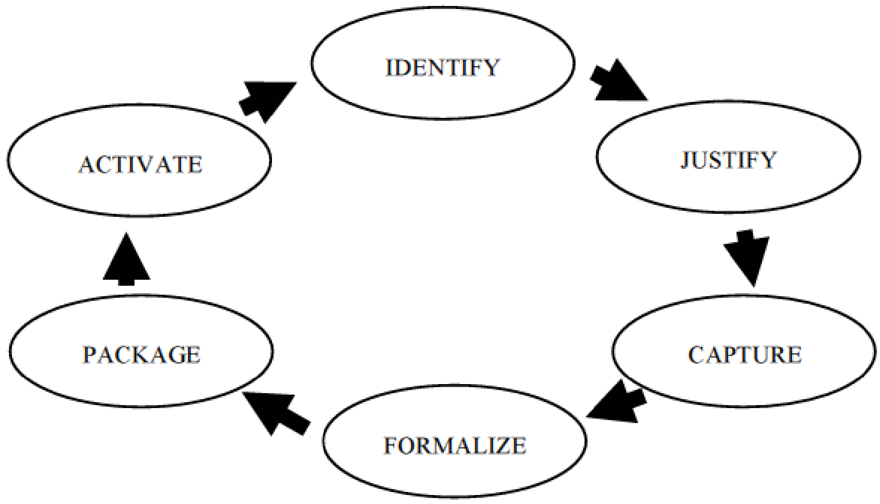


Figure 2. 8: MOKA Lifecycle (MOKA Consortium, 2001)

Identify – this stage takes existing business opportunity into account and selects the appropriate KBE application with resources needs necessary to build.

Justify – at this stage, technical, financial and cultural risk analysis is carried out by senior managers based on project plan developed to consider whether the project is viable or not.

Capture – raw knowledge is collected at this stage and structured into 'Informal Models' using ICARE (Illustration, Constraint, Activity, Rules and Entity) forms. ICARE elements are represented using process flow models and hierarchical charts.

Formalize – at this stage, the informal models are translated into formal models using MOKA Modelling Language (MML) which is based on the Universal Modelling Language (UML), making it easy for software engineers to understand.

Package – at this stage, the formal model is used to create actual software tools implemented in a KBE system.

Activate – at this stage, the software is installed, distributed and support made available for the daily use of the KBE system.

2.4.2.2 Intelligent Computer Aided Design (ICAD)

ICAD was the first CAD software in the field of KBE system developed in the early 1980s by KTI which increased the use of industrial design automation. ICAD has two interfaces, the first of which takes care of actual product geometry and the second interface handles programming of rules linked to the geometry. The system uses its own developed language known as the ICAD Design Language (IDL) which is based on a List Processing Language (LISP).

According to (Knutson, 2003) and (Bernard, 2003), the history of ICAD has been through a lot of changes which includes the sale of KTI to Dassault Systemes in 2002. ICAD in its earlier versions was made up of CAD engine with tools for capturing geometric knowledge and tool rules to generate 3D models automatically having specific input parameters. Cooper (2001) concludes that the ICAD system was extensively used within the automotive and aerospace engineering domain with major customers including Jaguar, British Aerospace, Airbus and others where Jaguar Motorcar Company applied it to assess manufacturing feasibility of vehicle headlights. However, more recent versions had an integrated approach, mainly centring on the knowledge and aspects of application development, and proving interfaces with common tools such as Parasolid, Solidworks, CATIA V5 and AutoCAD as well as with other standard desktop applications such as Excel (KTI, 2002).

2.4.2.3 Model-based and Incremental Knowledge Engineering (MIKE)

MIKE which stands for Model-based and Incremental Knowledge Engineering is a Knowledge Based Engineering methodology which supports iterative system development and prototyping compared with others which finalises all models before specific implementation begins (Studer, 1998). Angele (1998) mentions that MIKE “proposes the integration of semiformal and formal specification techniques and prototyping into an engineering framework”.

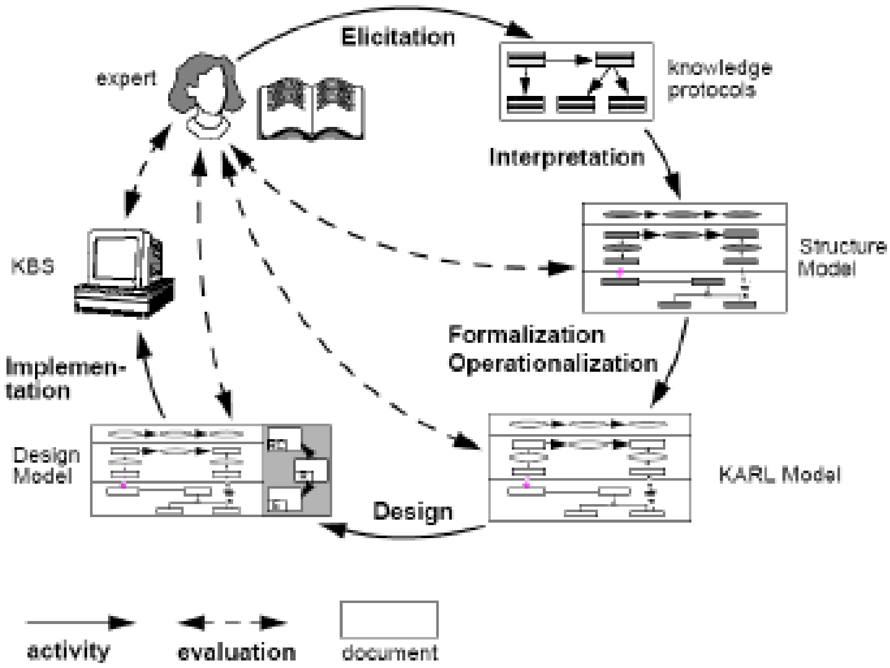


Figure 2.9: Main processes and deliverables of the MIKE methodology (Studer, 1998)

The different models in the MIKE methodology (Figure 2.9) shows various outcomes of different steps of the building process. The methodology claims that the knowledge engineers are responsible for representing knowledge in the elicitation model, structure model, model of expertise, design model and the final system. They either represent the same knowledge in different manners or extend existing models in another representation (Decker et. al., 1996).

According to Angele (1998) and Studer (1998), MIKE consists of two knowledge models, informal model and formal model. The informal knowledge model is created using techniques such as data flow diagrams, state transition diagrams, entity relationship diagrams and process flow diagrams which are high level representations of knowledge that is easily understood by users. However, the formal model is based on the capability of modelling in CommonKADS using Knowledge Acquisition and Representation Language (KARL). The KARL allows a working model to be tested in the entire development process by practically ensuring the completeness competence of the formal knowledge model. CommonKADS is a comprehensive methodology for developing knowledge based systems (Schreiber et al. 94).

Modelling in KARL requires the creation of an organizational model (de Hoog et al. 94) which considers the dynamic aspects, such as workflow, only in one direction. However, there is no description of how to integrate and use these models (Schreiber et al. 94).

2.4.3 Applications of Knowledge Based Engineering

Since the introduction of the concept, KBE has being used in diverse industries such as the automotive, aerospace, medical devices production, dental implants, commercial building systems, shipbuilding and many others.

Within the automotive industry, Chapman et al. (2001) used KBE to develop rapid car design system. Others also used the approach to within the automotive domain to design intelligent CAD systems [Georgia et al. (2009); Gardan and Gardan (2003); Deng et al. (1998); Berndt et al. (2009)]. However, Kumar et al. [2001; 2004; 2007; 2012; 2014] extensively used KBE for designing, selection and the reduction of cost of various dies. Furthermore, Tsuen Lin et al. (2012) applied Knowledge based Engineering for designing stamping dies using Functional-Based Stack-up Design System in CATIA system.

Tammineni et al. (2009) used a knowledge-based system for cost modelling of aircraft gas turbines for representing cost information of a design using a hybrid of hierarchical trees and object-oriented knowledge representation. The research led to the development of a tool which is able to provide incremental cost fluctuations in response to changes in component geometry. This research achieved very useful outcomes but limited to the aerospace industry and also users have limited chance in interrogating the manufacturing systems model which is behind the cost engine (Agyepong-Kodua et al., 2014)

Emberey et al. (2007) applied KBE in the aerospace engineering domain in support of engineering design applications development. In the same domain, Choi et al. (2005; 2007; 2009) extended the use of KBE concept for weight and estimation of cost. Also, Corallo et al. (2009) used KBE for the design of a low pressure turbine. The concept was also used by La Rocca et al. (2007) for automating aircraft wing body design.

In the domain of shipbuilding, Wu et al. (2011), Yang et al. (2012) and Arendt et al. (2011) have developed methods for ship design applying KBE. Cui et al. (2013)

Application of KBE in other domains includes research by Lee (2006) for a safety control system; Sawhney et al (2009) for web-based collaborative design system and Mourtzis et al. (2014), Wang et al. (2003) and Chen et al. (2003), for products design. for material selection research, Kumar et al.(2007, 2005), Sapuan (2001) and İpek et al. (2013) applied Knowledge Based Engineering approach.

2.5 New Product Introduction

New Product Introduction (NPI) could be a daunting task in the 21st century with the rise in organizational competition to meet and exceed customer specifications in terms of quality, functionality, fast delivery and flexibility. Where customers wants more for their money invested, enterprises are also to ensure that activities carried out to realize their customer’s requirements are cost effective. Having the right balance of customer expectation and cost incurred is critical to a firm’s success. The principle behind Levitt's (1960) treatise (“every declining industry was once a growth industry”) on competitive survival which necessitates the development of new products to replace current ones is still unchanged although the scope, direction of change and pace has increased dramatically over the years. Some products are developed with attention paid to consumer. (Hoffman, et al 2010; Fuchs, et al 2010), the nature of marketing venue (Fuller et al. 2009) (Arakji & K.R. 2007), the source of the product concept (Wyld 2010), the development process (Cooper 2009), the nature of the product (Decker & Scholz 2010) and the development process (Cooper 2009). Castellion & Markham (2013) observed success reported for product innovation research over the years at the rate 35 percent to 75 percent with their recent study reporting a consistent average of 40%. Talay et al. (2014) concludes that the success rate for NPI has to be improved, looking at the fact that it takes $1billion to develop and launch a new car model from scratch. Brondoni (2009) describes NPI as the process used by an organization to identify, design, create and bring to market new products or services. NPI is defined by the Product Development & Management Association based in USA as “A disciplined and defined set of tasks and steps that describe the normal means by which a company repetitively converts embryonic ideas into saleable products or services” (Belliveau et al. 2002). Most definitions of NPI according to Fixson (2009) include phases including idea generation, concept generation, opportunity identification, market and user analysis, concept refinement and selection, industrial design, prototyping, testing, financial evaluation and market introduction.

2.5.1 New Product Introduction Models

Models for an NPI process in most cases are applied to innovative products and processes. An innovative process is defined by (Gust-Bardon 2012) as a series of sequential changes, linked causatively, constituting stages of development of innovation. An innovative process was earlier described by (Niedzielski 2008) a sequence of events that has to take place in order to introduce a novel product to a market. However, (Gust-Bardon 2012) categorized NPI models into linear models and interactive models.

2.5.1.1 Linear Models

Kotler and Armstrong’s Model

According to (Kotler & Armstrong 2013), a systematic building of a new product development is a must for companies when introducing a good new product to the market. They defined a *new product* as original products, product improvements, product modifications, and new brands developed by organizations as a result of their own Research and Development efforts. Their research shows that NPI consists of eight major stages which are: idea generation; idea screening, concept development and testing; marketing strategy development; business analysis, product development; test marketing, and commercialization. Figure 2.14 shows the sequence of NPI process.

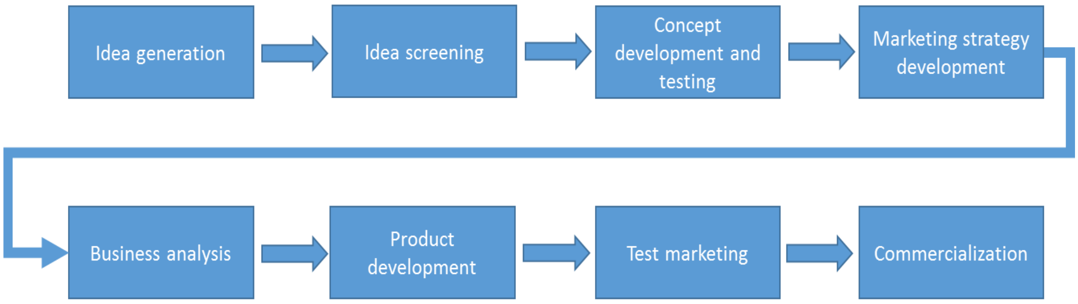


Figure 2. 10: Major Stages in New Product Development (Kotler & Armstrong 2012)

Idea Generation – this is the means of systematically searching for new product ideas. This process may result in a company generating hundreds or thousands of new product ideas in order to select few useful ones. New product ideas may be sourced mainly from internal customers (R & D efforts, manufacturing staff, executives, scientists, engineers and salespeople) or External customers (competitors, customers, distributors and suppliers). Galanakis et al. (2006) concludes that a company's creativity capabilities are necessary to their ability to innovate and survive in today's competitive environment.

Idea Screening – this stage reduces the number of ideas generated by spotting good ones and dropping poor ideas as soon as possible. As NPI cost rises later in the phases, only products that have the potentials of making a profit are carried on.

Concept Development and Testing – here, the product idea is detailed verbally or in a pictorial form in meaningful consumer terms to check the consumer's perception of an actual or potential product. In this sense, a targeted consumer group are involved to test the concept to assess the strength of consumer appeal (Fong 2003).

Marketing Strategy Development – initial marketing strategy for introducing an accepted product concept to the market is developed. The Marketing Strategy statement is made of three parts. The first describes the target market; then the planned product positioning and finally the sales, market share, and profit goals for the first few years (Kotler & Armstrong 2012).

Business Analysis – this stage analysis sales, costs, and profit projections for a new product concept to ensure that values are satisfactory to the company's objectives. Other experts (Nikolaos et al. 2004; Sandmeier et al. 2010) concludes that at this stage, decisions on the technical feasibility of the product, the products market potential and ultimately, the products financial contribution to the company are critically assessed.

Product Development – here, a physical product and several product prototypes are developed by R&D or engineering to ensure that the product idea can be turned into a workable product. The product is rigorously tested to ensure safety, efficiency, functional features and also the intended psychological characteristics.

Test Marketing – the developed product prototype and marketing program are introduced into realistic market settings. This allows marketers to initially test the product with customers before investing in full product introduction.

*Commercializatio*n – here, the new product is introduced into the market through the use of marking promotional tools. At this stage, the new product could be distributed intensively, exclusively or selectively to customers (Amue & Adiele 2012).

Stage Gate Model

Cooper (1990) identifies product innovation as a process and like other processes, introducing a new concept can be managed (Figure 2.11). His Stage-Gate system is an application of process-management methodologies to an NPI process. Stage-Gate is said to be ground-breaking and widely implemented by many experts, integrating it with numerous performance driving best practices to form an easy-to-understand recipe for success. Results for using Stage-gate are superior products reaching markets faster and more profitability (Edgett 2015). According to Nader et al. (2009), the Stage-gate concept consists of series of stages and gates where a stage is a specific research task and a gate is a checkpoint for decision making based on set criteria.

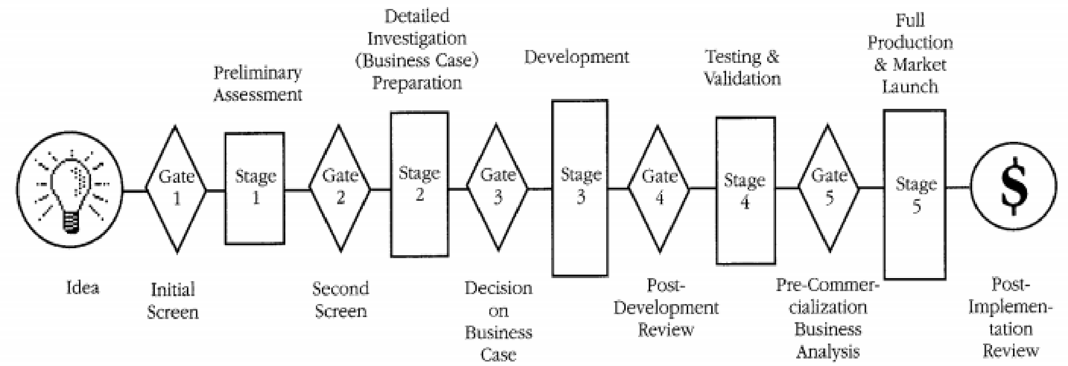


Figure 2. 11: An Overview of a Stage-Gate System (Cooper 1990)

Stages: Edgett (2015) confirmed that a typical Stage-Gate contains 5 stages in addition to a robust front-end or Idea stage which are:

Idea generation – business opportunities are discovered and uncovered and new ideas are generated.

Stage 1: Scoping – a quick desk research is carried out to preliminarily investigate and scope the project or product idea. This is usually inexpensive.

Stage 2: Build the Business Case – at this stage, primary research and detailed investigations on potential customers, market and technicalities on the product, leading to a Business case that includes product and project definition, project justification, and the proposed plan for development.

Stage 3: Development – at this stage, the actual detailed design of the product is developed as well as its operations and manufacturing processes required for full production. Prototypes are produced for testing.

Stage 4: Testing and Validation – pilot or batch production of the new products are tested at this stage in the lab, plant and marketplace for verification and validation of the new product and the production plans.

Stage 5: Launch (Commercialisation) – this stage marks the start full-scale production of the new product, marketing and sales.

Gates: Each stage has a gate where decisions are made whether or not to make further investment in the project (a Go/Kill decision). The gates serve as quality control checkpoints where three goals are to be achieved; ensuring the quality of execution, evaluating business rationale, and approving the project plan and resources. The gates have different purposes that they fulfil. For instance, Gate 1 activities are gentle for early screening of new product ideas whereas Gate 3 is tougher in terms of decisions on the business rationale for approving projects into a further expensive development stage.

The functionalities of the Gates are similar and they are structured in the following manner:

Deliverables – decision makers are known as Gatekeepers are assigned by the project leader and the team responsible for high-level results of completed activities during the previous stage.

Decision Criteria – decision criteria are set for every project to measure a clearly robustly defined success criteria capable of identifying winning products.

Output – a decision is made whether to go ahead with a project, hold on to the project, end the project or recycle. Where a project has a green light to continue, NPI resources are committed to the project and action plans approved for the next stage. A date for the next gate meeting is also set with a list of deliverables.

2.5.1.2 Interactive Models

According to (Gust-Bardon 2012), interactive models concentrate on the inputs and the interactions between networks. This modelling approach has gained popularity in literature over the last decade on networks open innovation and in lead user involvement (Pittaway et al. 2004)(Von Hippel 2005).

Coupling Model

The coupling model introduced by (Rothwell & Zegveld 1985) represents an interactive model as shown in Figure 2.112. According to them, the model is a logical and sequential but may not necessarily be a continues process but can be divided into a series of independent and yet interacting stages. The coupling model matches a company's technological capabilities with market needs at early stage of an innovative process. An innovative process was defined by them as a complex set of communication paths which are within and outside an organisation – which links a company with a wider scientific and technological community as well as with a broader market.

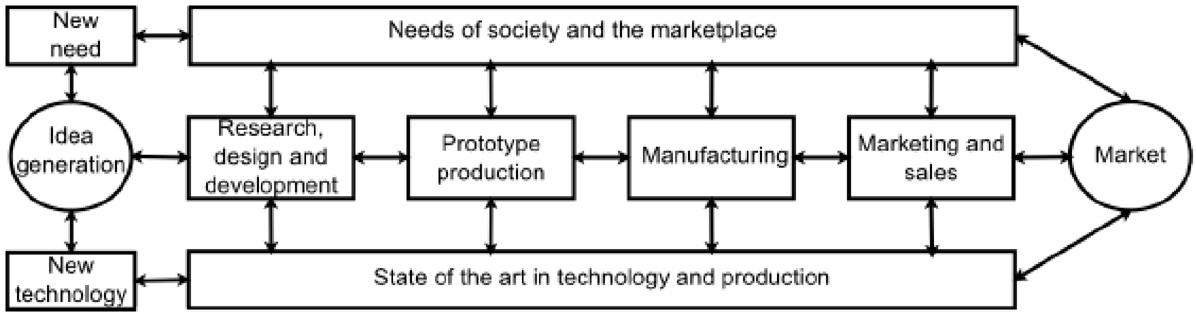


Figure 2. 12: The coupling model of an innovation process (Rothwell & Zegveld 1985)

Systematic Models

Systematic models gained attention through the research of (Freeman 1987; Lundvall 1992; Nelson 1993). The concept of the systematic models assumes two characters of innovation; social and evolutionary character of innovation. The social character considers the learning process which is the main activity of the model which simulates the interactions between people. According to (Freeman 1987), an innovative system refers to public and private institutions whose activities and mutual relations lead to creation, absorption, improvement and diffusion of new technologies. Innovation system was also defined by (Lundvall 1992) as elements and relationships that affect the development, diffusion and use of new, economically useful knowledge. (Nelson 1993) concluded that an innovative system is a group of institutions which has mutual interactions where the interactions affect the innovation performance of national companies.

Chain-Linked Model

(Kline & Rosenberg 1986) developed the chain linked model which consists of five stages and five paths of an innovation process as shown in Figure 2.12. According to their model, stage one identifies needs based on a potential market. Stage two deals with the invention and a production of an analytic design. In the stage, three combines detailed designs and testing whereas stage four modifies a project and then releases it for production. Finally, stage five distributes and market an innovation.

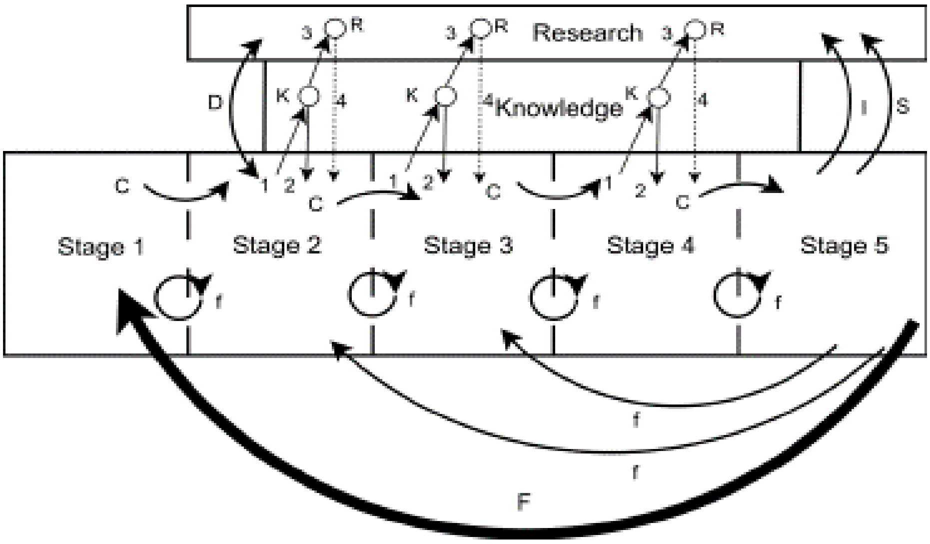


Figure 2. 13: The chain-linked model (Kline & Rosenberg 1986)

The five paths of the model are shown in Figure 2.13. as: C, f and F, K-R, D, I and S. Path C is known as the central chain of innovation consists of five stages which starts with the production of analytic design and ends with the production and marketing of an innovation. *f* and F are feedbacks that occurs between adjacent stages of an innovation process. Path K-R which is the third path connects research development (D) with innovation relying on available sources of knowledge. Path D shows the relationship between new research knowledge and how it leads to the emergence of a radical innovation (D). Finally, the path I and S shows how research is impacted (I) by innovative products as well as monitoring external developments through supporting the research (S).

2.6 Summary

Seven main areas in manufacturing systems engineering were discussed in this chapter. Section 2.1 introduced engineering cost estimation, where current cost estimation approaches in the research domain were reviewed looking at the various cost estimation techniques. Section 2.2 defined product and features, highlighting on how features are recognized, techniques for feature recognition and how features are represented in in the area of manufacturing systems. Section 2.3 looked at welding technologies, mainly focused on resistance spot welding, remote laser welding, gap control techniques in welding and also reviewed of various stitch parameters in welding. Furthermore, section 2.4 reviewed the literature on knowledge based engineering (KBE) where various definitions were reviewed as well as state of the art KBE methodologies and its application. Enterprise modelling research was reviewed in section 2.5, where the emphasis was placed on enterprise reference architecture and reference architecture methodologies. Finally, section 2.6 looked into the NPI process where NPI models were reviewed focusing on their benefits and limitations.

From the literature reviews, various gaps were identified in the area of manufacturing systems engineering. This research proposes solutions to gaps identified in cost estimation models (Table 2.2) and System Modelling and Cost Estimation Techniques (Table 2.3).

Table 2. 4: Related Literature Survey of Cost Estimation Models

Literature Criteria / **Cost Estimation Techniques**	**Approach**	**Gaps**	**Related Work**
Activity-Based Cost Estimation	It is applicable where all activities exist for a process. It uses existing resource data to calculate cost centre rates.	• Requires detailed data • More complex to allocate cost, hence expert knowledge needed. • It is expensive to implement and operate in an entire company.	Cooper and Kaplan (1987); Asiedu and GU (1998); Park and Kim (1995); (Lewis, 1993); (Turney, 1991); (Niazi et al., 2006); (Tornberg et al., 2002); (David Ben-Arieh, 2003); (Roy, 2003); Mikko, V., Marko, S., et al. (2007); Yazdifar et al. (2010) Wang, Du et al. (2010).

Parametric Cost Estimation	Uses Cost Estimating Relations (CER) based on Historic data by determining a correlation between dependent variable cost and independent variables.	• CER cannot be used for a new component or a novel technology with no available historical data. • Requires expert judgement for data acquisition and cost assessment.	Wright (1936);(Crawford, 1985); Curran et al. (2004); Agyapong-Kodua and Weston (2011); Agyapong-Kodua, et al. (2011); ISPA (2008); (Shepperd and Cartwright, 2001); (Akintoye and Fitzgerald, 2000); Younossi, et al. (2002).
Feature Based Cost estimation	It derives cost functions from existing objects linked to the engineering domain.	• There is no widely accepted definition of a feature across organization or industry. • Implications of this technique are not yet completely understood.	Rush and Roy (2000); Wierda (1991); Taylor (1998); Niazi, Dai et al. (2006); (Tiago Pascoal Filomena and Ez, 2011);
An Integrated P-P-R-Production Cost Estimation Technique	***Proposed in this thesis***		A technique that integrates Product-Process-Resource design cost with Production cost values for cost analysis on engineering changes.

Table 2. 5: Related Literature on System Modelling and Cost Estimation Techniques

Literature Criteria / **Enterprise Modelling and Engineering Cost Modelling**	**Approach**	**Gaps**	**Related Works**

Enterprise Modelling Frameworks • **PERA** • **GERAM** • **CIMOSA** • **TOGAF** • **VFF**	- focuses on general tasks - High level process description - Considers the entire lifecycle of an enterprise.	- May not accurately represent the process. - Does not show simultaneous and overlap, or concurrent - Identifying, gathering, and maintaining knowledge is a challenge - Process details are not captured.	Agyapong-Kodua et al., (2009); Kosanke, (1995); TOGAF (2011); Azevedo and Almeida, (2011); Hilton & P (2016)
Engineering Cost Modelling Techniques • **Qualitative** • **Quantitative**	Qualitative- Compares new product with a similar old to generate cost estimate. Quantitative – detailed analysis of product design	- Past design and manufacturing data is required to generate a reliable cost estimate. - Current cost modelling techniques are useful towards the final design phase where much data is available. - Interactions between new technological capabilities and emerging societal needs are not fully exploited.	Niazi, Dai et al. (2006); Westkämper et al., (2005); Berkhout, et al., 2010; Berkhout, et al., 2011; van der Duin, P., 2007; Berkhout et al., 2010; Francis, 2008; Chesborough’s (2004); (Huston and Sakkab, 2006; Thomke and Von Hippel, 2002; Vanhaverbeke and Cloodt, 2006

		- Does not emphasis on product design cost at early stages.	
PPR Modelling Technique	***Proposed in this thesis***		• A Product-Process-Resource Modelling Technique for Capturing Engineering Knowledge and Cost Values • A Technique for Extending Cost Modeller Capabilities to Include A New Process For Cost Assessment.

CHAPTER 3

PRODUCT-PROCESS-RESOURCE (PPR) COST ESTIMATION FRAMEWORK

Most academic research thesis and reports include research methodology which is an important aspect a research that has to be tackled carefully in great details, highlighting the various methodological approaches available and justifying the suitability of a chosen methodology or a combination. At this point, (Burton, 2002) observed that many PhD's Methodology chapters are a list of data collection and data analysis methods. According to (Kothari, 2004) research usually consists of *"defining and redefining problems, formulating hypothesis or suggested solutions; collecting, organising and evaluating data; making deductions and reaching conclusions and finally, carefully testing the conclusion to determine whether they fit the formulating hypothesis*". To justify the usefulness of a proposed research approach, (Collis and Hussey, 2003) concluded that research must be used for review or synthesize existing knowledge, investigating existing situations or problems, providing solutions to problems, exploring and analyse more general issues, constructing or creating new procedures or systems, explaining new phenomenon, generating new knowledge or a combination of any or all of the purposes above.

The proposed technique is scalable in its use in modelling, cost analysis and estimation in support of engineering decision making. There are three approaches to the proposed research methodology; firstly, it models data that captures engineering knowledge and extracts cost information for estimating product (P), process (P) and resources (R) design cost during early design stages. Engineering knowledge in this context refers to an understanding of engineering processes and resources that are consumed or expected to be consumed to realise a particular product or features of a product. Secondly, it demonstrates a systematic approach to executing static process flow model and resource flow models and resource database developed using commercial of the shelf tools such as Microsoft Visio into a cost modeller to extend its existing cost modeller capabilities. This solution is based on identifying cost modeller requirements and implementing compatible product, process and resource models that satisfy the requirements. Thirdly, a unique cost model that integrates product-process-resource design cost with production cost has been developed using cost accounting algorithms, to capture cost value changes made to product, process, resource or the manufacturing process parameters. This allows easy traceability of cost during design and production process.

3.1 Introduction to PPR Cost Estimation Framework

Product and process optimization sometimes require changes to product(s) features such as dimensions, materials, tolerances, shapes and so forth; equipment (new/modified tools and fixtures, robots, end effectors); new process; software installation and training to mention a few. These, however, may have a great impact on cost. For product designers to know the causalities of engineering changes, the author suggests a systematic approach to introducing cost as a key performance indicator (KPI) for innovative product designs. The proposed methodology integrates Product-Process-Resource models with a cost modeller to help with knowledge capturing and cost implication on engineering decisions at various stages of design. The novel concept is coined PPR Cost Estimation Framework. The algorithm behind this methodology is that product features can be associated with 'process capabilities' which can also be associated with 'resource competencies and capacities'. Cost is therefore generated through the consumption of 'resources' in the realization of 'processes'. As a result, changes to material selection for the product, processes or resource designs and their utilization have a significant causal impact on cost.

As discussed in the research scope in Section 1.5, the PPR Cost Estimation Framework consists of three major techniques:

1. A Product-Process-Resource Modelling Technique for Capturing Engineering Knowledge and Cost Values
2. A Technique for Extending Cost Modeller Capabilities to Include a New Process for Cost Assessment
3. An Integrated Cost Estimation Technique

As shown in Figure 3.1. PPR Cost Estimation framework also has the following iterative steps:

- ***3D CAD*** – this is given as input to the methodology in the form of a concept, idea, product prototype, product CAD model, etc. This is obtained from product designers.
- ***Data Modelling*** – develop graphical illustration, computer representation and cost estimation algorithms for Product-Process-Resource
- ***PPR Design Cost Calculator*** – integrate the cost estimation algorithms for Product-Process-Resource to assess the cost of initial designs.
- ***Decision Point 1*** – decision is taken based on cost values obtained against expected PPR design cost budget.
- ***Models Execution*** – developing a dedicated Virtual Production Environment (VPE) and integrating Process and Resources models into a cost engine of a cost modeller for estimating new manufacturing process cost.
- ***Decision Point 2*** – to check whether data available is enough to be implemented in cost modeller to be run successfully.
- ***Cost Assessment*** – Cost of Product is assessed based on data computed Process and Resources into cost Modeller.
- ***Integrated Cost Estimation Dashboard*** – this is a dashboard that shows the total cost of both design and manufacturing. This integrates the cost values obtained from the cost modeller and the developed PPR design cost model values.
- ***Decision Point 3-*** this is to benchmark cost values generated against budgeted cost values. Alternatives are also assessed at this point to reach acceptable cost values.
- ***Generate Report*** – Report is generated once cost information is satisfying to support decision making.

The initial assumption of the proposed methodology is that the 3D design of the product with its components are given as an input in a computer aided design (CAD) format. The framework contains iteration loops which allow more data to be added for better cost estimates. The data modelling stage requires the collection, structuring and modelling Product, Process and Resource data in a way that its output meets the input requirements of a cost modeller. It is also assumed that cost modellers cannot automatically predict the cost of using a new process as operations contained in the process and resources that have to be assigned are not yet known. The model execution stage extends existing cost modeller capabilities to include new processes and resources. Cost is then assessed based on the use of an alternative resource within the process as well as alternative materials to reach a reasonable cost value. Reports are then generated based on parameters changed to support decision making based on cost.

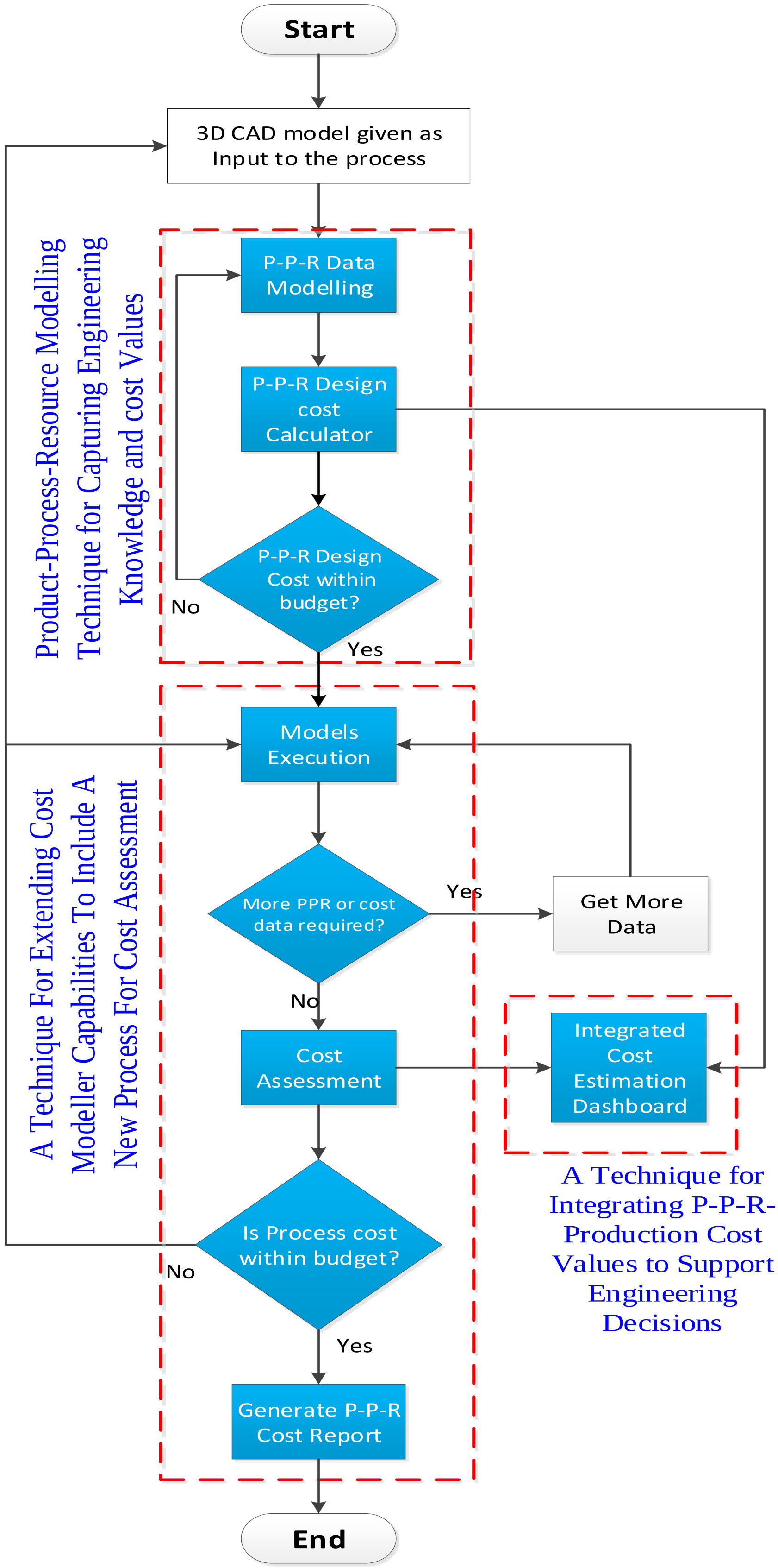


Figure 3.1: PPR Cost Estimation Framework

3.2 A Product-Process-Resource Modelling Technique for Capturing Engineering Knowledge and Cost Values

A new technique for capturing engineering knowledge for a new and innovative process and extracting cost values from developed models is proposed. The need for this technique is that, the current state of the art modelling techniques attempts to address the issue of the lack of an integrated engineering and cost modelling technique during early design stages. This section addresses the state of the art techniques. Identifies current research limitations and proposes a technique that addresses the limitation.

3.2.1 Engineering Modelling Techniques

Recent engineering activities have received enormous support through the advancement of IT. This has led to the development of a number of digital modelling tools. Currently, designers are supported by product design and analysis technologies, process assessment tools and resource performance and modelling technologies (Agyepong et al., 2011). These techniques tend to replicate real products and have a great impact on company finances. This is particularly so in the automotive and aerospace industry. It could be argued that the cost of purchasing a digital manufacturing (DM) software and training employees to use them is high but on the other hand, the benefits of its proper application override the cost. Marinov (2000) confirmed that DM software has the potential to predict problems at the design stage of a product's lifecycle before getting to the manufacturing phase, hence, avoiding the traditional and costly trial and error design approach and at the same time cutting down on designing, developing and manufacturing time to market a product. DM is able to describe aspects of the design-to-manufacturing process digitally using tools that support digital design, Computer-aided design (CAD), office documents, Product lifecycle management (PLM) systems, analysis software, simulation, Computer-aided manufacturing (CAM), etc. Chryssolouris et al (2009) indicated that DM is capable of initially shortening development time and cost; integrates knowledge coming from different manufacturing activities, processes and departments; decentralize manufacturing of increase in part or product variety in various production sites; finally focusing of enterprises on their core competencies, working efficiently with other companies and suppliers effectively using information technology. Academic literature reveals that DM is grouped into two categories; bottom-up methodology which considers the extension of DM concepts with a broader framework such as digital factory or enterprise and the other approach is a top-down which takes into account technologies that support individual activities which may include e-collaboration and simulation (Chryssolouris et al., 2009).

These achievements have led to dedicated and specialised tools for the engineering of different phases of the design process. Recent modelling advances in view of competitive engineering design outcomes have led to the notion of a 'digital factory' which is used to refer to a network of digital models, methodologies and applications required for the realisation and integration of activities within the full life cycle of a factory (Maropoulos & Ceglarek, 2010; Pedrazzoli, Sacco, Jönsson, & Boër, 2007; Tolio, Sacco, Terkaj, & Urgo, 2013; Yoon, Shin, & Suh, 2011). The main objective behind the digital factory model is that perceived operations and controls in a factory can be modelled and experimented until expected key performance indicators such as cost, quality and lead time are favourable (K. Agyapong-Kodua et al., 2009; Agyapong-Kodua, Haraszkó, & Németh, 2014; Hibino, Inukai, & Fukuda, 2006; Negahban & Smith, 2014). A number of researchers (Maropoulos & Ceglarek, 2010; Pedrazzoli et al., 2007; Tolio et al., 2013; Yoon et al., 2011) have reported significant benefits obtained through the digital factory concept.

3.2.2 Cost Modelling Techniques

An earlier study by Boehm (Boehm, 1984) revealed 7 cost modelling techniques with the potential to support engineering design analysis at an early stage of the product development cycle. Boehm (Boehm, 1984) identified the techniques as: Parametric, Expert judgment, Analogy, Parkinson, Price to Win, Top down and Bottom-up. The application of these techniques as reported in the earlier works of Boehm (Boehm, 1984) did not show how product design changes can impact directly on cost. This is crucial because the viability of an engineering decision must be carefully balanced with the economics of it. Later on, authors such as (K Agyapong-Kodua, J.O Ajaefobi, R.H Weston, & S Ratchev, 2012; Agyapong-Kodua, Asare, et al., 2014; Antonio C. Caputo & Pacifico M. Pelagagge, 2008; Seo, Park, Jang, & Wallace, 2002; E. Shehab & Abdalla, 2002; Tammineni et al., 2009) provided alternative cost modelling classifications when comparing industrially applicable techniques with techniques only common in academia. Some of the techniques identified were intuitive, parametric, variant-based, statistical, analogous, generative, analytical and feature-based. It was however identified that these published techniques were not directly associated with specific application needs leaving the user with a vast number of techniques to choose from. Also despite the different techniques reported, the present authors identified that there was limited knowledge on how the integrated strength based on the unified application of a set of cost modelling techniques can be harnessed. Shehab and Abdalla (E. M. Shehab & Abdalla, 2001) also showed that the predominant cost modelling techniques are intuitive, parametric, variant-based and generative. Other classifications are based on the methods for estimation- qualitative or quantitative approaches (Foussier, 2006; Layer, Brine, Van Houten, Kals, & Haasis, 2002). Qualitative methods usually adopt expert judgement and heuristic rules (Antonio C Caputo & Pacifico M Pelagagge, 2008). Although qualitative cost estimation methods cannot be overruled, Caputo (Antonio C Caputo & Pacifico M Pelagagge, 2008) argued that effort should be concentrated on quantitative cost modelling techniques. This is because qualitative methods only indicate whether an alternative is better or worse without absolute values (Agyapong-Kodua et al., 2011). An initial attempt to place a structure around cost modelling techniques was proposed by Agyapong-Kodua (Agyapong-Kodua, 2009b). The author proposed the use of system dynamic causal loops to outlay modelling needs and then based on cost modelling requirements, mapped the individual strength of the techniques unto the modelling requirements. Although this approach seems worthwhile, the author and his colleagues (K. Agyapong-Kodua, J. O. Ajaefobi, et al., 2012) upon further research concluded that no single solution is a panacea for successful cost modelling and there is the need for synergistic application of various techniques towards cost modelling. Although the proposed methodology is not a 'one stop' solution to the afore mentioned problem, it is a better alternative for capturing early design cost, due to its capability of integrating system modelling techniques with a cost modeller (aPriori). The proposed technique is an increamental improvement which helps solve this particular problem. Other researchers (Cavalieri et al., 2004; R. Curran et al., 2004) have shown that well described statistical models can help identify causalities and correlate cost and product characteristics to obtain a parametric function with one or more variables. This seems worthwhile and can help establish the correlation between product, process and resource variables. However, it has been carefully observed that currently, cost modelling has not been applied at a very detailed level in most companies except for major OEMs which have invested heavily into cost engineering projects hence have tools and experts to support their engineering project activities (Duverlie and Castelain, 1999; Layer et al., 2002; Matthews, 1983; Otswald, 1992; Stewart, 1991). Even in these cases, although the models are helpful at the design stage, the models lack information on cost composition and process plans which leaves designers less room to make adjustments to reduce cost. This is because usually when quoting a part or product, using just the cost parameter is not sufficient for the negotiation of the cost/delay ratio with the customer (H'mida, et al., 2006). Collopy and Curran (Collopy & Curran, 2005) reported that generally, there are cost modelling challenges associated with the complexity of cost, cost model validation, the presence of cost drivers outside designs and non-objectivity of estimates in some cases.

3.2.3 Limitations of The State Of The Art Techniques

Despite the success reported, more critically, although these models support engineering activities, detailed engineering analysis has not been closely integrated with economic analysis. Hence a viable engineering decision cannot currently be readily assessed economically. Currently, engineering analysis and economic analysis belong to two separate disciplines (Agyapong-Kodua et al., 2014). This is seen in the scope of this research and the conclusion section. As a result of this gap, the design life cycle can be longer than it should and there is the tendency of incurring avoidable errors (Rush and Roy, 2000). Basically, cost modelling is trailing behind engineering models.

3.2.4 Proposed Product-Process-Resource Data Modelling Technique

Based on the above identified research gaps, a new modelling technique that integrates systems modelling techniques with cost models is proposed. With this technique, product (P), process (P) and resources (R) data are modelled using business process modelling notation (BPMN) in SysML for illustration. A computer code is then generated for the developed models as a computer representation in an Extensible Markup Language (XML) that is both human-readable and machine-readable. In addition, a cost model is developed using standard cost accounting equations.

The PPR data Modelling stage shows the data transformation process of the framework. This process requires the modelling of:

- Product Data
- Process Data
- Resource Data
- Process and Resource Integration Data

Each model consists of the following three elements:

1. *Illustration* (graphical model-based) – the use of system modelling tools and techniques to capture and visualise the flow of product, process and resources.

2. *Computer Representation* (script) – developing scripts from the product, process and resource flow models that are readable to cost modellers and for other engineering simulation purposes.

3. *Cost Estimation Algorithms* – creating cost equations by combining cost accounting techniques with manufacturing cost estimation techniques.

The input required at this stage of the proposed methodology is a complete 3D CAD model of a product with its individual components having geometric features. Based on the 3D CAD, models and databases are created capable of supporting cost values generation. The schematics of the PPR Modelling Stage is shown in Figure 3.2, showing the relationship between the models and the details contained in each model.

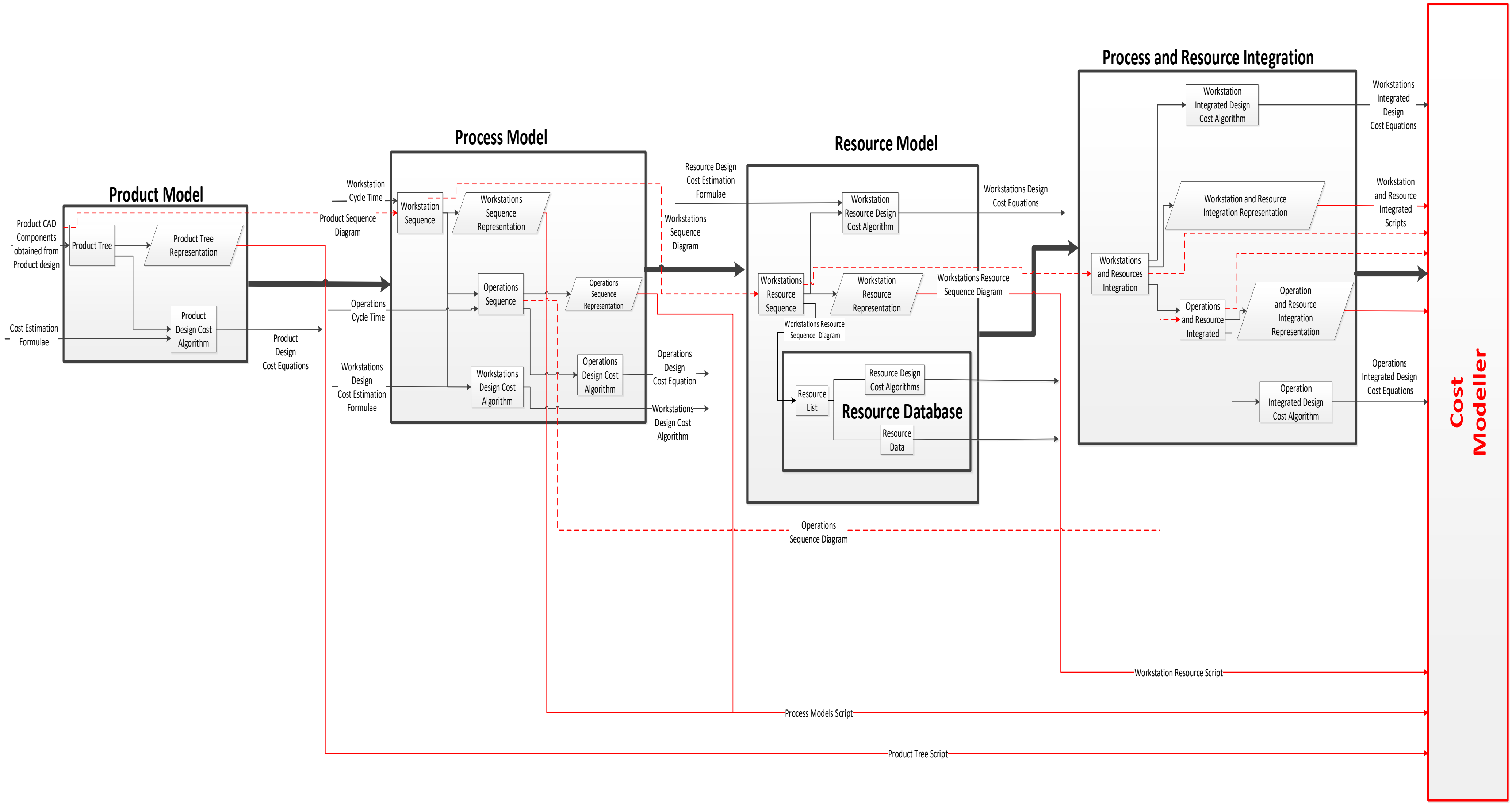


Figure 3.2: PPR Data Modelling Schematic

The benefits of creating these models in the methodology aids with the systematic capturing design knowledge required for estimating the cost of introducing a new product. Figure 3.3 shows a graphical flow summary of the PPR Data Modelling stage of the methodology.

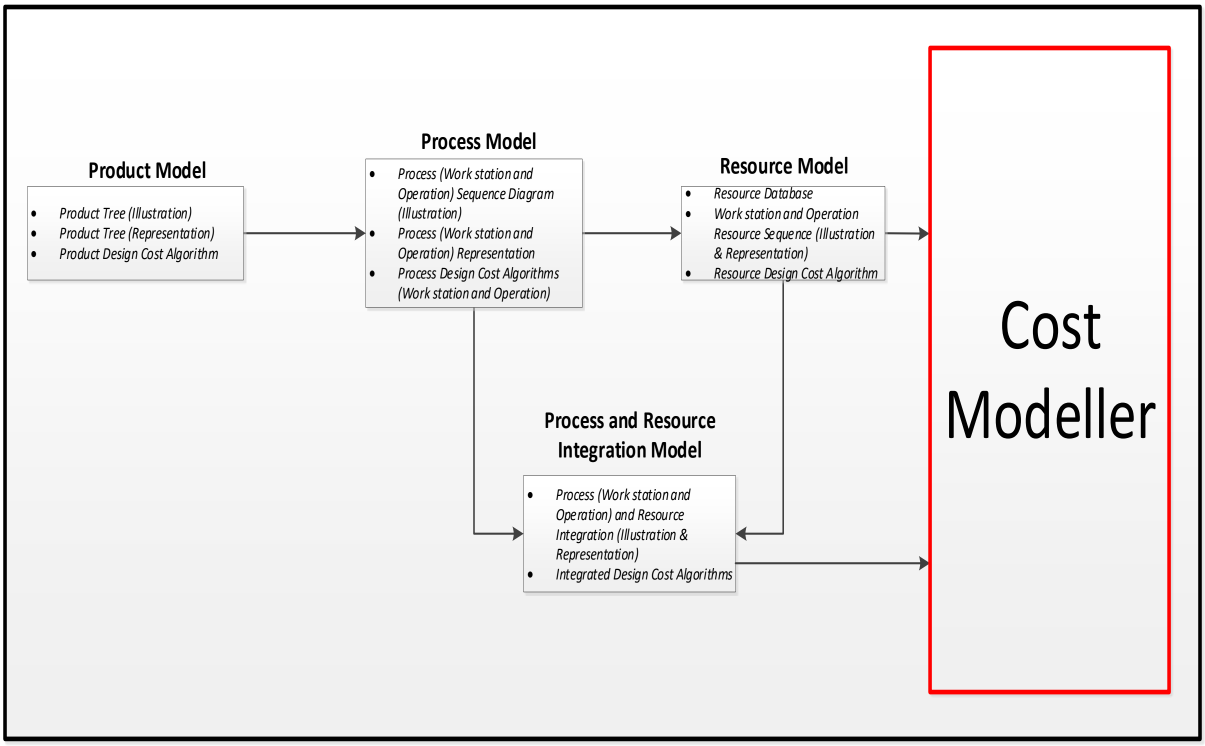


Figure 3.3: Summary of PPR Data Models

The data modelling schematics shows the content of each model and the relationship between the various models. This allows cost calculations to be done at the data modelling stage of the methodology as soon as information on Product, Process and Resources becomes available at the early design stage. This is to help capture the various process knowledge and cost information in real time during design stage as the design process mature to help with design decision making related to cost rather than waiting to complete the entire design and then calculate the cost. The models, therefore, contain all necessary cost algorithms for estimating cost at early design stage of an NPI process. Each model is further expanded in more detail in the later sections.

Processing the data is done for the following reasons;

1. To systematically capture the dynamics of Product, Process and Resources as their knowledge increases with time
2. To generate input parameters required by cost modeller for estimating cost at early stage of an NPI process where information is not complete
3. To estimate cost through various stages for an innovative product development.

Systems Modelling Landguage (SysML) is used for developing the Product model, Process model, Resource model and Process and Resource Integration model as sown in Section 3.2.4.1 to 3.2.4.4. SysML is a modelling language that is well understood and used by systems designers for static represention of systems. The logic behind this is to show a static representation of the Product (P), Process (P) and Resource(P) models of a system and then generate HTML scripts of the PPR models. The HTML is then used as an input to the cost modeller, where in this case application, aPriori cost modeller is used.

3.2.4.1 Product Model

Product Models has inputs, modules and expected outputs. As shown in Figure 3.4, creating the Product Models requires the creation of 3D models of all the product components, a product tree, a cost estimation calculation and computer representations.

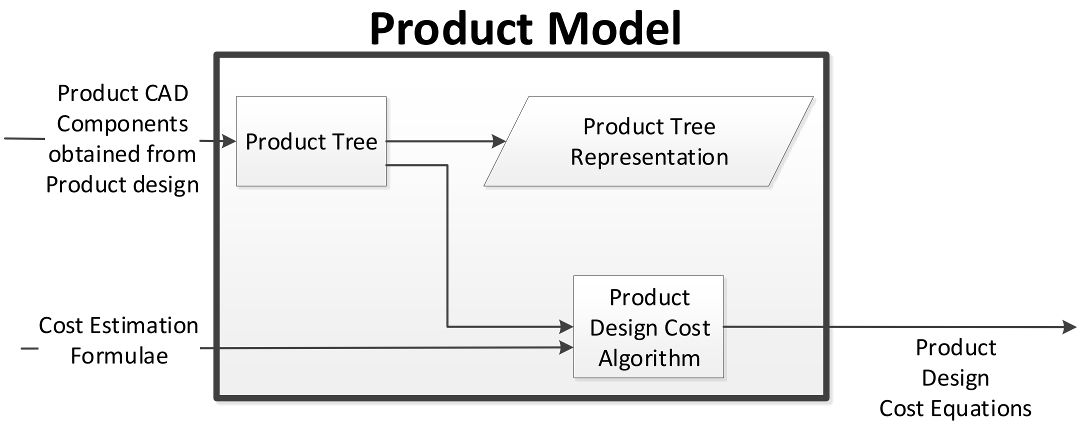


Figure 3.4: Product Models

(1) Product Model Inputs

These define what is required to generate product models for the PPR Cost Estimation Framework. This work is not limiting the inputs required for generating product models but suggesting generic feeds for innovative product design.

Product Requirement – geometric information with functionalities and performance of the product are captured or given to product designer for designing an innovative product. Although at the early design stage information may not be available, there have to be systems in place to support designers when information is required.

3D CAD Components**: -** traditional design and manufacturing processes assumes that 3D CAD models of Product(s) are given to Manufacturing as an input by the Product designer. However, this approach may not be valid for smaller companies where there are no separate design and manufacturing departments but rather one person or a group does both design and manufacturing. On the other hand, big companies with separate design and manufacturing departments may have experts users of supporting Product lifecycle management (PLM) tools. For example, with most CAD design tools, the CAD Products contain geometric and non-geometric information. However, Material information and manufacturing process for realising the Product may not be seamlessly integrated with product CAD models. There is therefore the need for process designers to develop a platform for manufacturing processes capable of realising CAD products.

For the purpose of this research, the 3D CAD model of the product is given as input to the proposed methodology and is assumed to be obtained from product design.

(2) Product Modules

a) ***Product Tree*** – the Product Tree shows how product components are sequentially assembled, the precedence and relationships with each other. From the product components received from product designer, a Product List is generated to ensure that product tree model can be easily understood. The tree may be created using a top-bottom or a bottom-up approach. For example, a typical product tree model is represented graphically in Figure 3.5 showing the final product having sub assembly A and B, with the sub assemblies components.

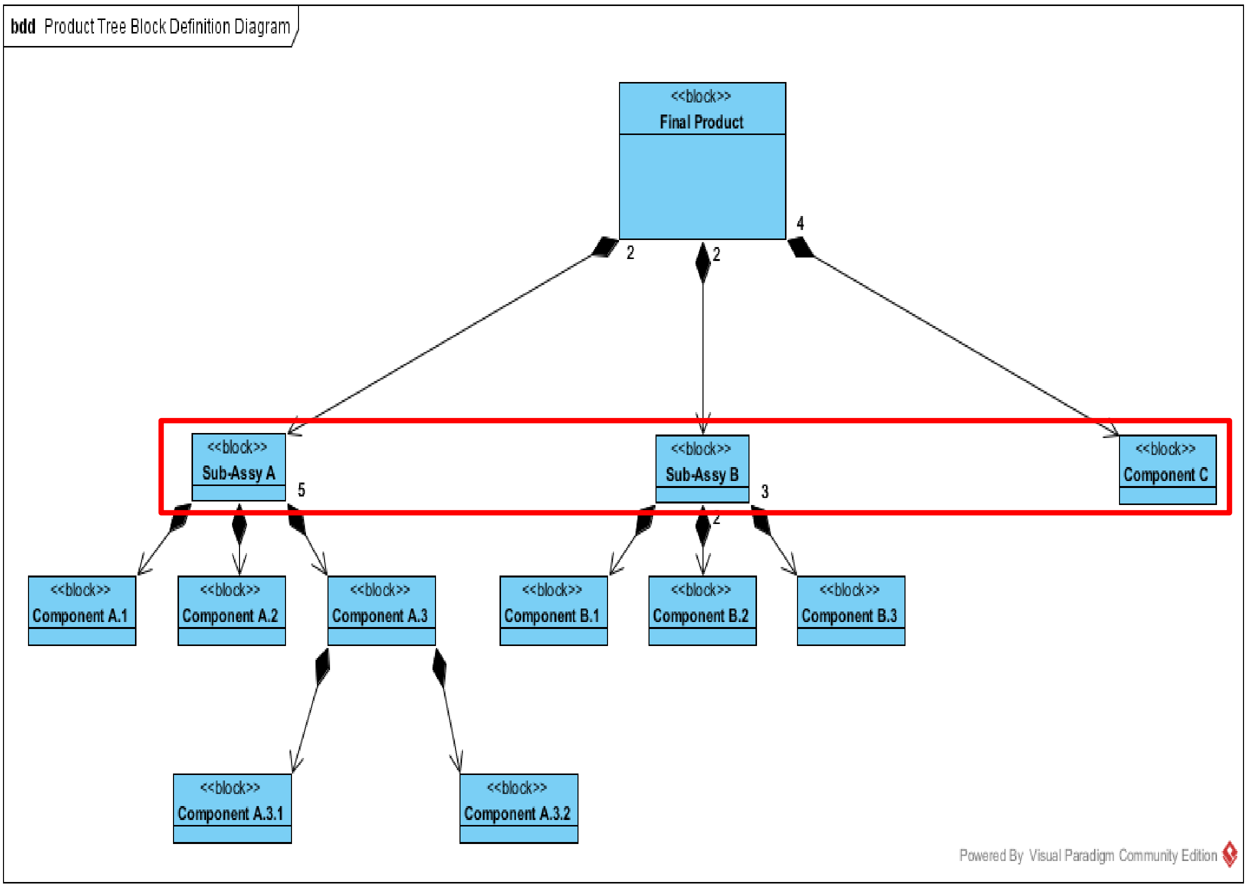


Figure 3.5: Product Tree Illustration

The Product Tree is created using SysML with a top-bottom approach. Other graphical illustration tools may also be used to represent the Product Tree.

The highlighted region of Figure 3.5 shows the sub assembly components of the final product. The representation of this in an XML format is also shown in Figure 3.6, where Sub-Assy A, Sub-Assy B and component C are also highlighted. This shows how the static models can be translated to a computer readable format.

b) ***Computer Representation:*** – a computer representation of the Product Tree is generated in a structured text and shown as in Figure 3.6 in an XLM format. This shows how the product tree created graphically in SysML in Figure 3.5 is translated into a computer representation for other computer systems applications. This representation is useful particularly for the modification and interrogation of existing cost estimation tools. A complete XML computer structured representation of all process and resource models created in SysML can be found in Appendix D.1.


```
PmLastModified="2016-10-03T14:34:02.386" PmCreateDateTime="2016-08-16T15:03:58.106" PmAuthor="asare_k" Documentation_plain="">
<Stereotype Name="block" Id="nWe3h.qESzVmxxA4" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-10-
  20T12:55:46.803" PmCreateDateTime="2016-06-29T15:11:25.625" PmAuthor="asare_k" Documentation_plain="" Leaf="false"
  Abstract="false" Root="false" IconPath="" BaseType="SysMLBlock"/>
+ <SysMLBlock Name="Final Product" Id="lWe3h.qESzVmxxA3" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-10-
20T12:55:46.803" PmCreateDateTime="2016-10-20T12:50:47.209" PmAuthor="asare_k" Documentation_plain="" Visibility="public" Leaf="false"
Abstract="false" ConnectToCodeModel="1" Root="false" Encapsulated="false" BusinessModel="false" BusinessKeyMutable="true" Active="false">
+ <SysMLBlock Name="Sub-Assy A" Id="EI0Ph.qESzVmxxBg" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-10-
20T13:05:47.029" PmCreateDateTime="2016-10-20T12:53:50.088" PmAuthor="asare_k" Documentation_plain="" Visibility="public" Leaf="false"
Abstract="false" ConnectToCodeModel="1" Root="false" Encapsulated="false" BusinessModel="false" BusinessKeyMutable="true" Active="false">
+ <SysMLBlock Name="Sub-Assy B" Id="WHiPh.qESzVmxxB." UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-10-
20T13:05:47.029" PmCreateDateTime="2016-10-20T12:54:18.010" PmAuthor="asare_k" Documentation_plain="" Visibility="public" Leaf="false"
Abstract="false" ConnectToCodeModel="1" Root="false" Encapsulated="false" BusinessModel="false" BusinessKeyMutable="true" Active="false">
+ <SysMLBlock Name="Component C" Id="W9GPh.qESzVmxxCa" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-10-
20T12:55:46.803" PmCreateDateTime="2016-10-20T12:54:46.106" PmAuthor="asare_k" Documentation_plain="" Visibility="public" Leaf="false"
Abstract="false" ConnectToCodeModel="1" Root="false" Encapsulated="false" BusinessModel="false" BusinessKeyMutable="true" Active="false">
+ <SysMLBlock Name="Component A.1" Id="DSRPh.qESzVmxxDX" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-10-
20T13:05:47.029" PmCreateDateTime="2016-10-20T12:55:25.232" PmAuthor="asare_k" Documentation_plain="" Visibility="public" Leaf="false"
Abstract="false" ConnectToCodeModel="1" Root="false" Encapsulated="false" BusinessModel="false" BusinessKeyMutable="true" Active="false">
+ <SysMLBlock Name="Component A.2" Id="lS1Ph.qESzVmxxD3" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-10-
20T13:05:47.029" PmCreateDateTime="2016-10-20T12:56:02.089" PmAuthor="asare_k" Documentation_plain="" Visibility="public" Leaf="false"
Abstract="false" ConnectToCodeModel="1" Root="false" Encapsulated="false" BusinessModel="false" BusinessKeyMutable="true" Active="false">
+ <SysMLBlock Name="Component A.3" Id="ZwDPh.qESzVmxxEV" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-10-
20T13:05:47.029" PmCreateDateTime="2016-10-20T12:56:21.606" PmAuthor="asare_k" Documentation_plain="" Visibility="public" Leaf="false"
Abstract="false" ConnectToCodeModel="1" Root="false" Encapsulated="false" BusinessModel="false" BusinessKeyMutable="true" Active="false">
+ <SysMLBlock Name="Component B.1" Id="p3PPh.qESzVmxxFF" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-10-
20T13:05:47.029" PmCreateDateTime="2016-10-20T12:57:14.341" PmAuthor="asare_k" Documentation_plain="" Visibility="public" Leaf="false"
Abstract="false" ConnectToCodeModel="1" Root="false" Encapsulated="false" BusinessModel="false" BusinessKeyMutable="true" Active="false">
+ <SysMLBlock Name="Component B.2" Id="Mvgvh.qESzVmxxFj" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-10-
20T13:05:47.029" PmCreateDateTime="2016-10-20T12:57:34.924" PmAuthor="asare_k" Documentation_plain="" Visibility="public" Leaf="false"
Abstract="false" ConnectToCodeModel="1" Root="false" Encapsulated="false" BusinessModel="false" BusinessKeyMutable="true" Active="false">
+ <SysMLBlock Name="Component B.3" Id="nRovh.qESzVmxxGF" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-10-
20T13:05:47.029" PmCreateDateTime="2016-10-20T12:57:49.625" PmAuthor="asare_k" Documentation_plain="" Visibility="public" Leaf="false"
Abstract="false" ConnectToCodeModel="1" Root="false" Encapsulated="false" BusinessModel="false" BusinessKeyMutable="true" Active="false">
+ <SysMLBlock Name="Component A.3.1" Id="Srivh.qESzVmxxGx" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-10-
20T13:05:47.029" PmCreateDateTime="2016-10-20T12:58:39.954" PmAuthor="asare_k" Documentation_plain="" Visibility="public" Leaf="false"
Abstract="false" ConnectToCodeModel="1" Root="false" Encapsulated="false" BusinessModel="false" BusinessKeyMutable="true" Active="false">
+ <SysMLBlock Name="Component A.3.2" Id="rDmvh.qESzVmxxHP" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-10-
20T13:05:47.014" PmCreateDateTime="2016-10-20T12:59:12.437" PmAuthor="asare_k" Documentation_plain="" Visibility="public" Leaf="false"
Abstract="false" ConnectToCodeModel="1" Root="false" Encapsulated="false" BusinessModel="false" BusinessKeyMutable="true" Active="false">
</Models>
+ <Diagrams>
</Project>
```


Figure 3.6: Snap Shot of Product Tree Script Representation

c) ***Product Design Cost Estimation Algorithm: -*** estimating the cost of an innovative product through its design process is the core of the PPR Cost Estimation Framework. For the purpose of demonstrating the methodology, a simplified cost model is developed and shown (Eq 3.1 to 3.7) using designer's rate and time. However the final model created is scalable which may include other costs such as software licence cost, training cost, prototyping costs, consultancy costs and other overhead costs. Cost equations are created at various stages of the product design to support engineers with cost information necessary for engineering decision making. The cost estimation algorithm, however, generates the cost incurred in the designing of the master 3D CAD models of all required components, taking into account all necessary iterations expected during the design process. The algorithm for estimating the cost of product design depends on (1) number of components of the final product, (2) the design man hours expected to be spent on designing, (3) the hourly rate of designers and (4) the number of designers involved in designing. The function for the cost estimation algorithm is expressed as:

Cost Inputs

R_D = *Designers' Rate*

T_D = *Designer's Time*

N_D = *Number of Designers*

C_m = *Material Cost*

Product component design cost function is given as;

$$\sum_{k=1}^{n} C_{D,k}\ (C_{D,1} + C_{D,2} + C_{D,3} + \cdots C_{D,n}) \qquad (3.1)$$

Where;

$k = number\ of\ product\ components$

$C_{D,1};\ C_{D,2};\ C_{D,3};\ ...\ C_n$= Product component costs

C_D = Design cost component to be calculated

$$C_{D,k} = (R_{D,k} \times T_{D,k} \times N_{D,k}) \qquad (3.2)$$

Where

$T_{D,k} = Estimated\ Design\ Time$

$N_{D,k} = Number\ of\ designers$

$$R_{D,k} = \frac{Product\ Designer's\ Annual\ salary}{Predicted\ Annual\ Hours} \qquad (3.3)$$

Material Cost (C_m) is added as an option available for analysis purposes at the design stage for product designers

For example, the cost equation for estimating the cost of a product with three components may be expressed as; Component Design Cost$(C_D) = \sum(C_{D,1} + C_{D,2} + C_{D,3} + \ldots C_{D,n})$.

Product inspection cost is calculated using equation;

$$C_{INSP} = (R_{INSP} \times T_{INSP} \times N_{INSP}) \quad (3.4)$$

Where

$R_{INSP} = Inspector\ Rate$

$$R_{INSP} = \frac{Inspector's\ Annual\ salary}{Predicted\ Annual\ Hours} \quad (3.5)$$

$T_{INSP} = Estimated\ Inspection\ Time$

$N_{INSP} = Numbers\ of\ Inspectors$

$$Total\ C_{INSP} = \sum(C_{D,1(INSP)} + C_{D,2(INSP)} + C_{D,3(INSP)} + \cdots C_{D,n(INSP)}) \quad (3.6)$$

Where;

$(C_{D,1(INSP)};\ C_{D,2(INSP)};\ C_{D,3(INSP)};\ \ldots C_{D,n(INSP)})$

$= Cost\ of\ inspecting\ individual\ designed\ product\ components.$

Therefore, the equation for calculating the cost of designing a product is given as the sum of design cost and inspection cost which is given as;

$$C_{PROD} = \sum(\sum_{k=1}^{n} C_{D,k}\ (C_{D,1} + C_{D,2} + C_{D,3} + \cdots C_{D,n}) + (\sum(C_{D,1(INSP)} + C_{D,2(INSP)} + C_{D,3(INSP)} + \cdots C_{D,n(INSP)}) \quad (3.7)$$

(3) Product Model Outputs

The outputs of the Product Models generated are:

- *Product Sequence Diagram*
- *Product Tree Script*
- *Product Design Cost Equations.*

As shown in Figure 3.2, the product sequence diagram is in red arrow and it becomes an input to the Process Model. The product tree, on the other hand, is required later in the framework during the implementation stage of the cost modeller. Finally, the Product Design Cost Equations is used at the product model stage for estimating the cost of designing the product as well as for estimating the cost of making engineering changes

3.2.4.2 Process Model

Creating a process model is necessary and it helps in having an overview of activities within a system. A process model refers to the flow of activities from one workstation to the other done in a predefined order to realize a product. A process is defined in this research as a sequence of interdependent and linked operations which consumes resources. The process model is, therefore, a visual representation of how an enterprise is expected to function. The process model can be represented in static or dynamic models. Representation of a graphical process model may be obtained using software tools such as ARIS Express, Microsoft Visio, Process Maker, Visual Paradigm or Microsoft PowerPoint. Moreover, the PPR estimation approach requires the use of SysML or any process modelling tool or language that is capable of generating XML files for modelling visual static models. These are mainly artificial languages capable of expressing data, information, knowledge or a system consistently using a set of rules. The most common languages are Universal Modelling Language (UML), Systems Modelling Language (SysML), Integrated DEFinition Method 3 (IDEF3), Event Driven Process Chain (EPC), Petri Net and Role Activity Diagram (RAD). The Process Model as shown in Figure 3.7 breaks the process down to Workstation level and Operations level to help capture and understand the process required for realising the final product.

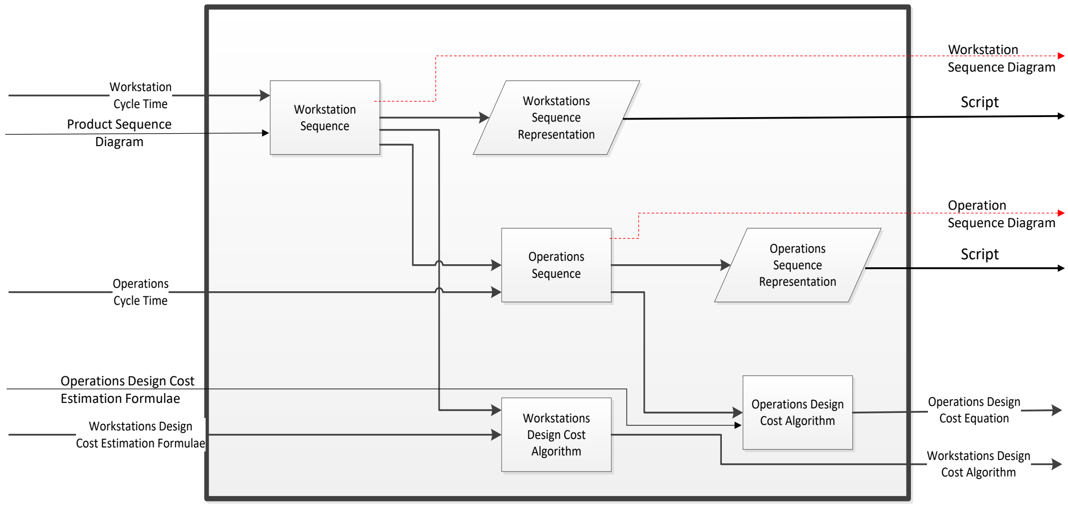


Figure 3. 7: Process Model

The Process Models consists of the following modules:

- Workstation Sequence
- Sequence Representation (Workstation and Operation)
- Design Cost Algorithm (Workstation and Operation)
- Operation Sequence

(1) Process Model Input

Product Sequence Diagram: - this input helps the process designer with the organization of workstations to ensure the smooth running of the process as well as for process optimization purposes.

Cycle Time: - an estimated Cycle Time for the overall realization of the product is determined. High level time estimate is apportioned to the various Workstations and then detailed Cycle Times are allocated to Operations within the workstation.

Design Cost Estimation Formulae: - the cost estimation formulae for the process determines the cost involved in designing the manufacturing process, which involves estimation formula for both Workstation and Operation level design.

(2) Process Modules

Process modules show a graphical representation of the Workstations and their interdependence with each other as well as the workstation's Operations.

Workstation Sequence: - Figure 3.8 is an example of a graphical representation of a system with relationship between various Workstations. In this example, the process is divided into Workstations 1-7 where workstations may represent departments or functions within the organization. Creating the sequence of the workstations helps with having an overview of the relationships between functions or departments.

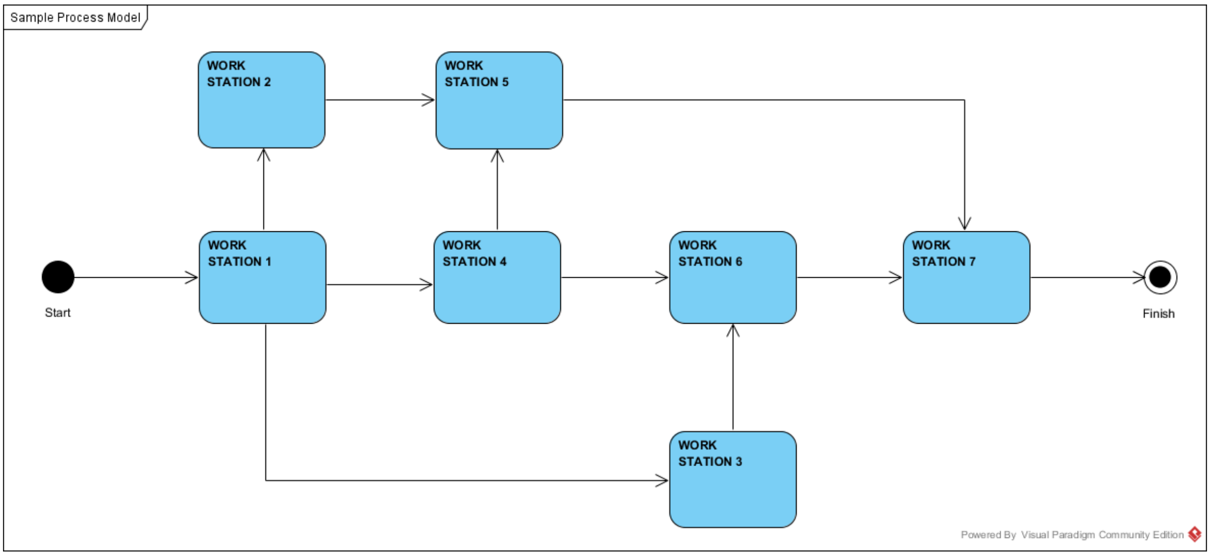


Figure 3.8: Sample Process Model

Operation Sequence: - Furthermore, each Workstation is decomposed into more detailed operational activities. This research defines the detailing of a function or a workstation within the process as an operation. For example, Work Station 2 in Figure 3.8 has various operational activities that have to be fulfilled in order to proceed to the next Workstation 5. The operational activities for Workstation 2 are illustrated in Figure 3.9. However, each operation may further be decomposed into sub-activities, however, this research limits decompositions to the operations level.

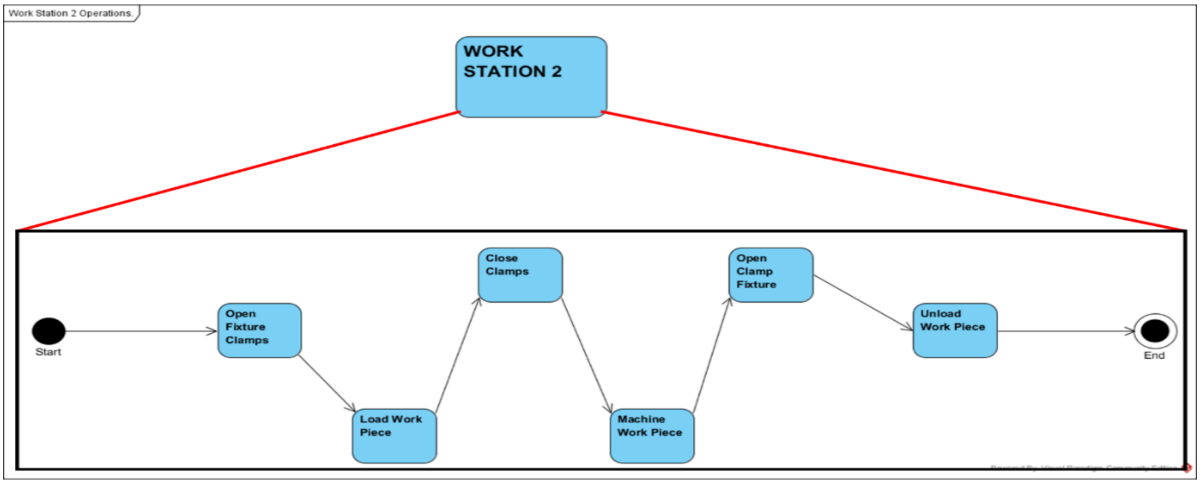


Figure 3.9: Workstation Model Flow

Sequence Representation: - A computer representation of both Workstation Sequence and Operation Sequence are generated as shown in Figures 3.10 and 3.11 in an XML format. These representations (a complete representation can be found in the link in Appendix D.1) shows how computer systems are able to utilise and interpret process models created in a different modelling design tool. These representations are useful for analysis of the modelled process in dynamic modelling system tools as well as for modification or optimization purposes of the existing model. The sequence representations are also useful for verification and validation of the created models.

Figure 3.10: Workstations Level Computer Representation

Figure 3.11: Operations Level Computer Representation

***Process Design Cost Algorithm:* -** similar to the product design cost calculation, Process Design Cost Calculation assists the process designer to have an overview of cost information at the early stages of the design process. Estimating the cost of a process handles it at the Workstation and Operations level. The sum of operation design cost makes the cost for its workstation and the total cost of all workstations gives the estimated Process Design Cost.

Cost Inputs

R_{WD} = Workstation Designer's rate

T_{WD} = Workstation Designer's Time

N_{WD} = Number of Workstation Designers

R_{OD} = Operations Designer's Rate

N_{OD} = Number of Operation Designers

T_{OD} = Operation Designer's Time

$$\text{Process Design Cost} = \sum \text{Workstation Design Cost} + \sum \textit{Workstations Inspection cost} \quad (3.8)$$

Workstation Design Cost = C_{WD}

Operations Design Cost = C_{OD}

$$C_{WD,j} = \sum_{j=1}^{n} (R_{WD,j} \times T_{WD,j} \times N_{WD,j}) \quad (3.9)$$

Where j = number of workstation = *1, 2, 3……, n*

$$R_{WD} = \frac{\textit{Workstation Designer's Annual salary}}{\textit{Predicted Annual Hours}} \quad (3.10)$$

$$C_{W(INSP)} = \sum_{j=1}^{n} (R_{W(INSP),j} \times T_{W(INSP),j} \times N_{W(INSP),j}) \quad (3.11)$$

Where:

$R_{W(INSP),j} = \textit{workstation design inspectors rate}$

$T_{W(INSP),j} = \textit{workstation design inspection time}$

$N_{W(INSP),j} = \textit{Number of workstaions to be designed}$

$$R_{W(INSP),j} = \frac{\textit{Workstation Inspector's Annual salary}}{\textit{Predicted Annual Inspection Hours}} \quad (3.12)$$

Therefore, Process Design Cost equation is given as:

Process Design Cost $(C_{PD,j})$

$$= \sum_{j=1}^{n} (R_{WD,j} \times T_{WD,j} \times N_{WD,j}) + \sum_{j=1}^{n} (R_{W(INSP),j} \times T_{W(INSP),j} \times N_{W(INSP),j}) \tag{3.13}$$

(3) Process Model Output

As shown in Figure 3.2., the outputs of the Process Model are:

- *Workstation sequence diagram*
- *Operations sequence diagram*
- *Workstation sequence script*
- *Operations sequence script*
- *Workstation Design Cost Estimation Equations*
- *Operation Design Cost Estimation Equations*

Workstation sequence diagram then becomes an input to the Resource Model Design.

3.2.4.3 Resource Model

Modelling Resource Model is necessary for having oversight of organising and determining resources that are expected to be consumed in workstations and its operations. A resource is defined in this research as machines, tools, equipment, materials or people that are consumed within a process for the realization of a product or service. Figure 3.12 displays the inputs, modules and outputs of the Resource Model.

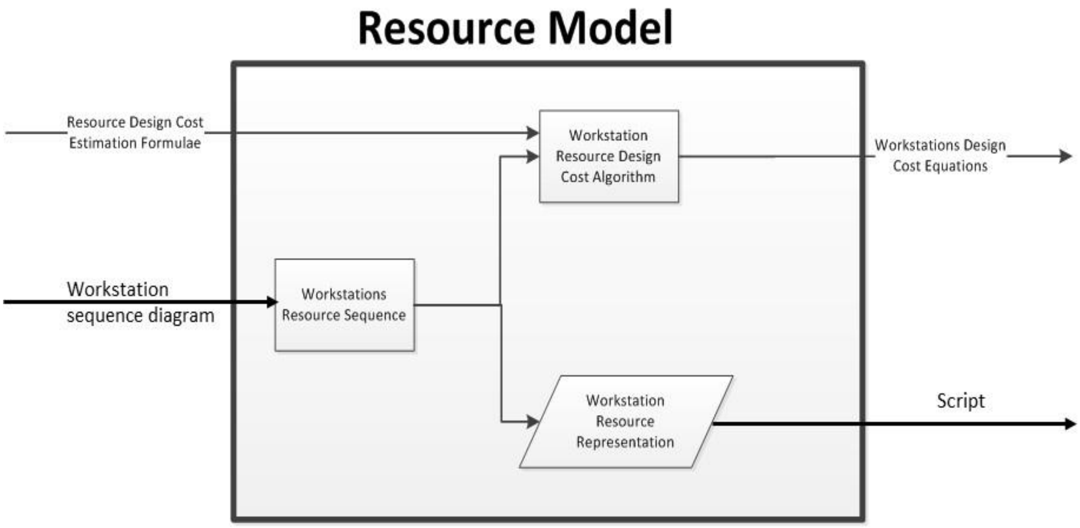


Figure 3.12: Resource Model

(1) Resource Model Inputs

Workstation Sequence Diagram**:** - the Resource Model takes in the workstation Sequence Diagram as input as if a system is being modelled at the workstation level. However, modelling workstation's operations will also require *Operations Sequence Diagram.*

(2) Resource Model Modules

***Resource Database* -** Creating a database of resources for a process necessary, particularly in a complex environment where cost is considered as a key performance indicator. At the early design stage of creating innovative products, there is less information on resources, hence, the use of tools such as Microsoft Excel and Access can be useful for organising and maintaining resource data as they become available. Creating a resource database requires modules such as Resource List Resource Capabilities and Resource Cost Information as shown in Figure 3.12. This makes it easier to capture all resources required to be used in the process. A simplified Resource Database created in Microsoft Excel is expressed in Table 3.1, where the database is scalable as resource list increases. Although table 3.1 shows a simplified version of the resource database, other factors, such as energy consumption, floor space utilisation, depreciation, maintenance cost, setup time, etc. values were also considered and computed in the cost estimator for the case application in chapter 4 of this research.

Table 3. 1: Sample of a Resource Database

Resource ID	Resource Description	Resource Cost (£)	Hourly Rate (Per Hour)	Quantity
Mil-1	Milling Maching	300,000	15	3
RB-10	Pick and Place Robot	500,000	20	6

Workstation Resource Sequence**:** - a resource list is initially generated with the aim of identifying resource needs for each workstation. Also, individual resources functionalities are modelled as shown in Figure 3.13.

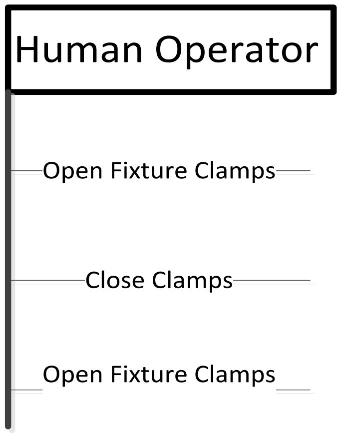


Figure 3.13: Example of a Human Resource Function

The resource sequence shows how resources within a process are networked with each other with their precedence and relationships. Figure 3.14 shows an example of a resource sequence for a process. The resource model is outputted from Systems Modelling Language (SysML), a tool that is generally understood by system designers for graphically representing processes and resource flow. The tool helps in showing the relationship between resources.

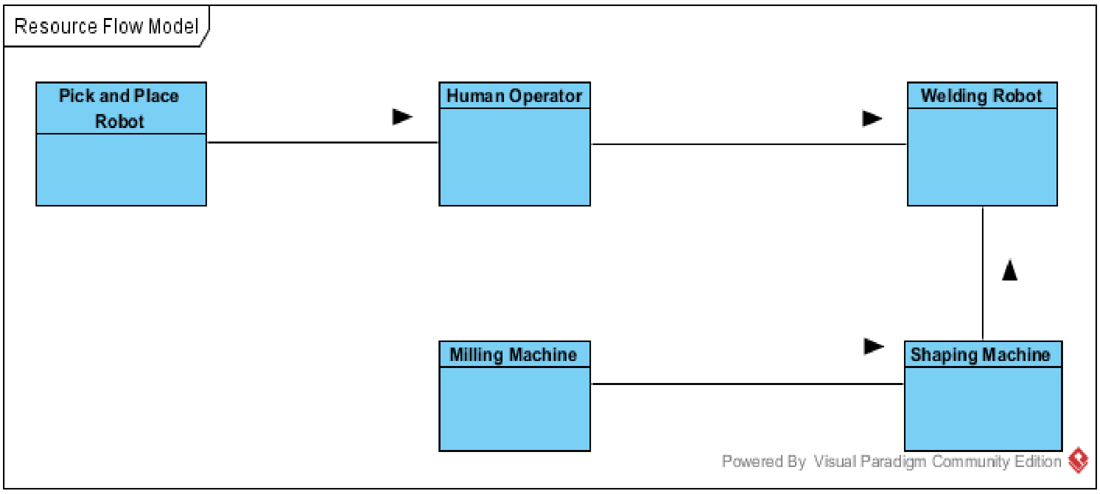


Figure 3.14: Resource Flow Illustration

The Resource Sequence also shows the sequence in which resources are to be introduced into a process to ensure the overall efficiency of the process. The black diamond on the lines shows the flow of resources in SysML. Obviously, some resources may be required earlier in a process, whereas some may be required later. Nevertheless, in some cases, some resources are utilized simultaneously. A computer representation of the resource flow is shown in Figure 3.15

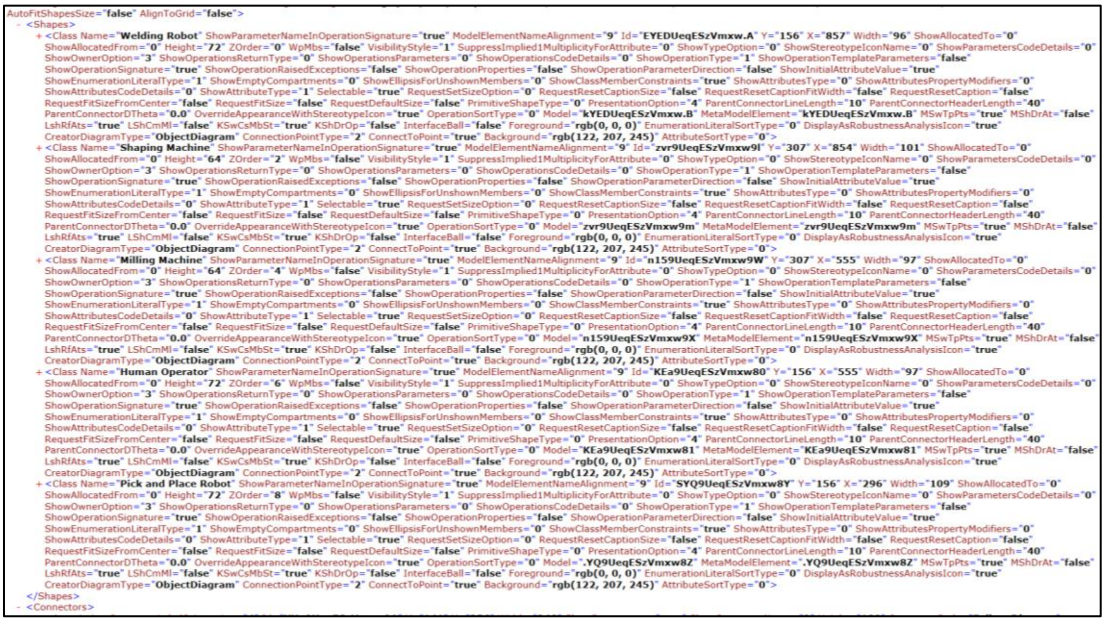


Figure 3.15: Resource Flow Representation

Resource Design Cost Algorithm: - this takes into account the cost of designing resource for a process as well as the cost of implementing those resources within the process. Estimating the cost of resources required to be consumed in a process requires the use of the following input parameters:

Resource Design Cost Inputs

R_{RD} = Resource Design Rate

T_{RD} = Resource Designers' Time

N_{RD} = Number of Resource Designers

C_{RD} = Resource Design Cost.

The cost equation for estimating the cost of designing a resource for a workstation is the sum of design and inspection costs. The resource design cost may be calculated using;

$$C_{RD,m} = \sum_{m\geq 1}^{n} (R_{RD} \times T_{RD}) \tag{3.14}$$

Where

m = number of Resource Designs

$T_{RD} = Time\ required\ for\ resource\ design$

$R_{RD} = Resource\ designer's\ rate$

$$R_{RD} = \frac{Annual\ Salary\ of\ Resource\ Designer}{Expected\ Annual\ Working\ Hours} \tag{3.15}$$

Inspection cost is also calculated using

$$C_{R,(INSP)} = \sum_{m\geq 1}^{n} (R_{RD,(INSP)} \times T_{RD,(INSP)}) \tag{3.16}$$

Where;

$R_{RD,(INSP)} = Resource\ inspector's\ rate$

$T_{RD} = Time\ required\ for\ resource\ inspection$

$$R_{RD,(INSP)} = \frac{Annual\ Salary\ of\ Resource\ Inspector}{Expected\ Annual\ Working\ Hours} \tag{3.17}$$

$$Total\ C_{RD,m} = \sum_{m\geq 1}^{n}(R_{RD} \times T_{RD}) + \sum_{m\geq 1}^{n}(R_{RD,(INSP)} \times T_{RD,(INSP)}) \tag{3.18}$$

Resource Sequence Representation*:* - a computer representation of the resource sequence is required for implementation in cost modeller. Expanding the capabilities of a cost modeller requires scripting or coding in a language that is compatible with the cost modelling system. In general, most systems work fine with HTML files as shown in Figure 3.16

```
<?xml version="1.0" encoding="UTF-8"?>
<Project Xml_structure="simple" UmlVersion="2.x"
TextualAnalysisHighlightOptionCaseSensitive="true" Name="Process Models"
ExporterVersion="12.2" ExportedFromDifferentName="false" DocumentationType="html"
Description="<html> <head> <style type="text/css"> <!-- body { color: #000000; font-
family: Dialog; font-size: 11px } --> </style> </head> <body> </body>< /html> "
CommentTableSortColumn="Date Time" CommentTableSortAscending="false"
Author="asare_k">

//this shows the association between resources used in the sample process model and the tasks
assigned to each resource//

<ModelChildren><Association Id="M..iUVqESzVmxw13" UserIDLastNumericValue="0"
QualityScore="-1" PmLastModified="2017-03-01T14:42:25.973"
PmCreateDateTime="2017-03-01T14:28:45.644" PmAuthor="asare_k"
Documentation_plain="" Visibility="Unspecified" OrderingInProfile="-1" Leaf="false"
EndRelationshipToMetaModelElement="s98iUVqESzVmxw0t"
EndRelationshipFromMetaModelElement="YhRCUVqESzVmxw0b" Direction="From To"
Derived="false" Abstract="false"><FromEnd><AssociationEnd Name="Move Part"
Id="M..iUVqESzVmxw14" UserIDLastNumericValue="0" QualityScore="-1"
PmCreateDateTime="2017-03-01T14:28:45.644" PmAuthor="asare_k"
Documentation_plain="" Visibility="Unspecified" Leaf="false" Derived="false"
TypeModifier="" Static="false" ReadOnly="false" ProvidePropertySetterMethod="false"
ProvidePropertyGetterMethod="false" Navigable="Unspecified"
Multiplicity="Unspecified" JavaCodeAttributeName=""
EndModelElement="YhRCUVqESzVmxw0b" DerivedUnion="false"
ConnectToCodeModel="1" AggregationKind="None"><Qualifier><Qualifier Name=""
Id="M..iUVqESzVmxw15" UserIDLastNumericValue="0" QualityScore="-1"
PmCreateDateTime="2017-03-01T14:28:45.644" PmAuthor="asare_k"
Documentation_plain=""/></Qualifier><Type><SysMLBlock Name="Human Operator"
Idref="YhRCUVqESzVmxw0b"/></Type></AssociationEnd></FromEnd><ToEnd><Associati
onEnd Id="M..iUVqESzVmxw16" UserIDLastNumericValue="0" QualityScore="-1"
PmCreateDateTime="2017-03-01T14:28:45.644" PmAuthor="asare_k"
Documentation_plain="" Visibility="Unspecified" Leaf="false" Derived="false"
TypeModifier="" Static="false" ReadOnly="false" ProvidePropertySetterMethod="false"
ProvidePropertyGetterMethod="false" Navigable="Navigable" Multiplicity="Unspecified"
JavaCodeAttributeName="" EndModelElement="s98iUVqESzVmxw0t"
DerivedUnion="false" ConnectToCodeModel="1"
AggregationKind="None"><Qualifier><Qualifier Name="" Id="M..iUVqESzVmxw17"
UserIDLastNumericValue="0" QualityScore="-1" PmCreateDateTime="2017-03-01T14:28:45.644"
PmAuthor="asare_k" Documentation_plain=""/></Qualifier><Type><SysMLBlock Name="Robot
1" Idref="s98iUVqESzVmxw0t"/></Type></AssociationEnd></ToEnd>
```


Figure 3.16: Computer representation of Resource Flow

(3) Resource Model Outputs

The Resource Model generates the following outputs:

- *Workstations Resource Sequence Diagram*
- *Workstations Resource Sequence Scripts* and
- *Resource Design Cost Equations.*

3.2.4.4 Process and Resource Integration

Integrating processes and resources is the next step of the PPR Cost Estimation Framework. Process flow by itself does not make much sense but Resources consumed within processes generates cost. The ability to show the relationship between processes and their corresponding resources helps with defining potential cost to be incurred in workstations as well as aiding the overall cost estimation calculations. The integration is done at two levels;

- the workstation level and
- the operation level

Figure 3.17, shows the modules for the Process and Resources Integration model for creating the workstation and operation levels of integration.

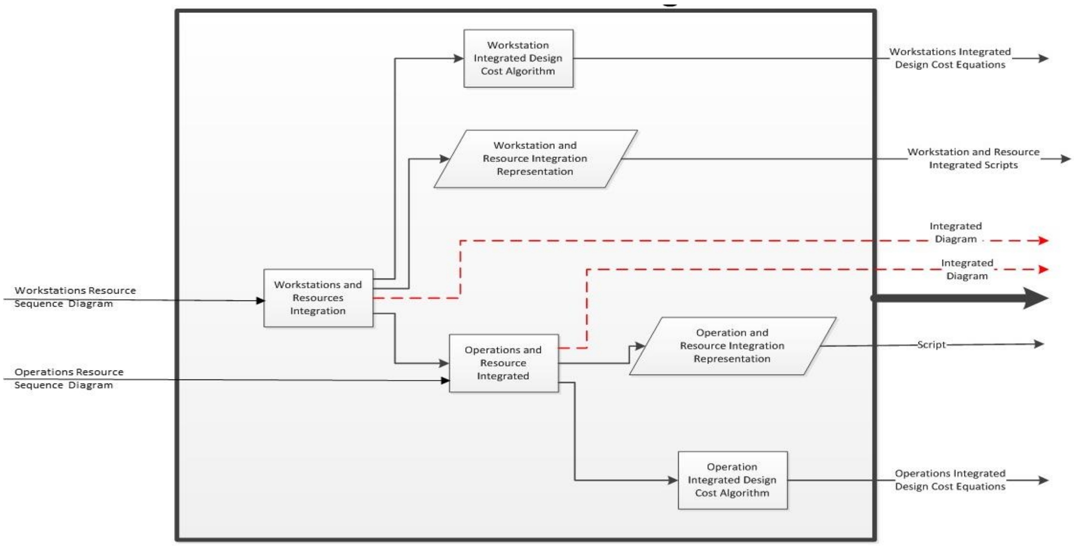


Figure 3.17: Process and Resource Integration Model

(1) Process and Resource Integration Inputs

Resource Sequence Diagram (Workstation)*:-* the Resource sequence diagram at workstation level is required for integrating and assigning resources to workstations within a process.

Resource Sequence Diagram (Operations)*:-* Resource Sequence diagram at operations level is required for assigning resources to workstations operations.

(2) Process and Resource Integration Modules

Workstations and Resource Integration:- Figure 3.18 shows how resources are assigned to workstations in a process represented in a use case. It is important at this stage that all workstations are allocated resources where in some instances, a particular resource may be allocated to more than one workstation or one workstation may have various resources allocated to it. The workstation level of integration shows a high level of resource engagement in a process environment. Resources ustilised in Figure 3.18 is highlighted in red and their relationship with the workstations are shown with the arrows.

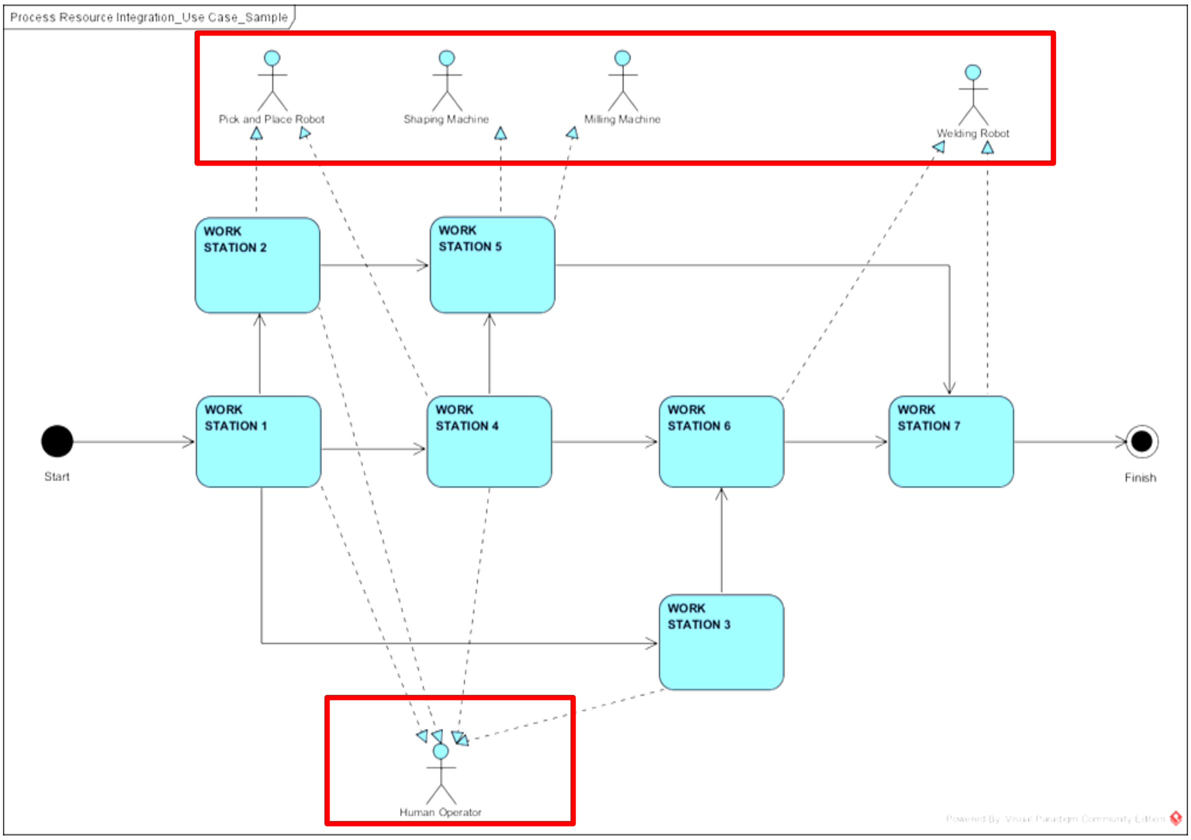


Figure 3.18: Workstation and Resource Integration Model

Workstations and Resource Integration Representation: - A computer representation of the workstation and Resource integration is then generated as shown in Figure 3.19. This illustrates how cost modeller systems interpret the workstation and resources integration in 'XML' format.

Figure 3.19: Computer Representation of Workstations and Resources Integration

***Operations and Resources Integration**: -* a more detailed integration is demonstrated at the operations level, where workstation activities are assigned. An example of this is illustrated in Figure 3.20.

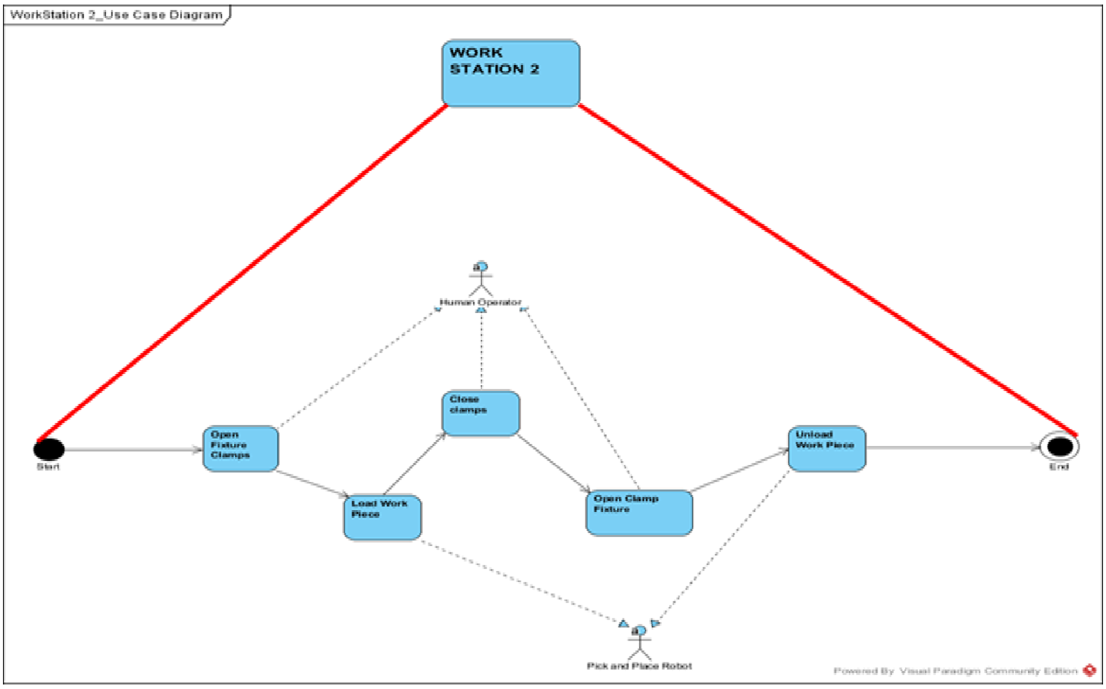


Figure 3.20: Operations and Resources Integration

The operational level of integration is to ensure that resources are assigned to all activities within an operation. It also gives an overview of the utilization of resources' capabilities.

Operations and Resource Integration Representation: - resources assigned to workstations operations are interpreted in a way that can be understood by most computer systems for other applications such as process optimization. Figure 3.21 shows how operations and resources are integrated in 'XML' format using SySML.


```
<Actor Name="Human Operator" Id="adoGqeqESzVmxxMQ" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-10-05T17:17:00.517" PmCreateDateTime="2016-10-05T16:05:02.614" PmAuthor="asare_k" Documentation_plain="" Visibility="public" Leaf="false" UserID="737e1d94-faa1-4618-8de3-ad414d22f3d1" Root="false" FontSize="0" BusinessModel="false" Abstract="false"><FromSimpleRelationships><Realization Name="" Idref="TjMmqeqESzVmxxQO"/><Realization Name="" Idref="TdCmqeqESzVmxxQr"/><Realization Name="" Idref="3nSmqeqESzVmxxQ8"/><Realization Name="" Idref="ThbWqeqESzVmxxTE"/><Realization Name="" Idref="PoM5qeqESzVmxxXc"/><Realization Name="" Idref="trc5qeqESzVmxxXr"/><Realization Name="" Idref="qgi5qeqESzVmxxX6"/></FromSimpleRelationships> <MasterView><Actor Name="Human Operator" Idref="adoGqeqESzVmxxMP"/></MasterView> </Actor><Activity Name="WORK STATION 2" Id="pFapqeqESzVmxxT4" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-10-05T17:06:41.435" PmCreateDateTime="2016-10-05T17:02:59.753" PmAuthor="asare_k" Documentation_plain="" SingleExecution="false" Reentrant="false" ReadOnly="false" Precondition="" Postcondition="" BodyFontSize="0"><MasterView><Activity Name="WORK STATION 2" Idref="JFapqeqESzVmxxT2"/></MasterView> </Activity><Activity Name="Open Fixture Clamps" Id="rFapqeqESzVmxxT9" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-10-05T17:17:00.519" PmCreateDateTime="2016-10-05T17:02:59.768" PmAuthor="asare_k" Documentation_plain="" SingleExecution="false" Reentrant="false" ReadOnly="false" Precondition="" Postcondition="" BodyFontSize="0"><FromSimpleRelationships><ControlFlow Name="" Idref="IVapqeqESzVmxxUl"/></FromSimpleRelationships> <ToSimpleRelationships><ControlFlow Name="" Idref="SVapqeqESzVmxxU0"/><Realization Name="" Idref="PoM5qeqESzVmxxXc"/></ToSimpleRelationships> <MasterView><Activity Name="Open Fixture Clamps" Idref="rFapqeqESzVmxxT7"/></MasterView> </Activity><Activity Name="Load Work Piece" Id="_FapqeqESzVmxxUC" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2017-02-14T17:09:26.265" PmCreateDateTime="2016-10-05T17:02:59.776" PmAuthor="asare_k" Documentation_plain="" SingleExecution="false" Reentrant="false" ReadOnly="false" Precondition="" Postcondition="" BodyFontSize="0"><FromSimpleRelationships><ControlFlow Name="" Idref="4VapqeqESzVmxxUq"/><ControlFlow Name="" Idref="2.mZvFqESzVmxwgc"/></FromSimpleRelationships> <ToSimpleRelationships><ControlFlow Name="" Idref="IVapqeqESzVmxxUl"/><Realization Name="" Idref="9rK5qeqESzVmxxYJ"/></ToSimpleRelationships> <MasterView><Activity Name="Load Work Piece" Idref="_FapqeqESzVmxxUA"/></MasterView> </Activity><Activity Name="Close clamps" Id="4lapqeqESzVmxxUH" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-10-05T17:17:00.519" PmCreateDateTime="2016-10-05T17:02:59.785" PmAuthor="asare_k" Documentation_plain="" SingleExecution="false" Reentrant="false" ReadOnly="false" Precondition="" Postcondition="" BodyFontSize="0"><FromSimpleRelationships><ControlFlow Name="" Idref="JqJ5qeqESzVmxxZO"/></FromSimpleRelationships> <ToSimpleRelationships><ControlFlow Name="" Idref="4VapqeqESzVmxxUq"/><Realization Name="" Idref="trc5qeqESzVmxxXr"/></ToSimpleRelationships> <MasterView><Activity Name="Close clamps" Idref="4lapqeqESzVmxxUF"/></MasterView> </Activity><InitialNode Name="Start" Id="GlapqeqESzVmxxUR" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-10-05T17:06:41.434" PmCreateDateTime="2016-10-05T17:02:59.801" PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" Leaf="false" MustIsolate="false"><FromSimpleRelationships><ControlFlow Name="" Idref="SVapqeqESzVmxxU0"/></FromSimpleRelationships> <MasterView><InitialNode Name="Start" Idref="6lapqeqESzVmxxUP"/></MasterView> </InitialNode><ActivityFinalNode Name="End" Id="ZlapqeqESzVmxxUW" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-10-05T17:06:41.433" PmCreateDateTime="2016-10-05T17:02:59.815" PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" Leaf="false" MustIsolate="false"><ToSimpleRelationships><ControlFlow Name="" Idref="hVapqeqESzVmxxVD"/></ToSimpleRelationships> <MasterView><ActivityFinalNode Name="End" Idref="plapqeqESzVmxxUU"/></MasterView> </ActivityFinalNode><Activity Name="Open Fixture Clamps" Id="1lapqeqESzVmxxUb" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2017-02-14T17:09:26.265" PmCreateDateTime="2016-10-05T17:02:59.820" PmAuthor="asare_k" Documentation_plain="" SingleExecution="false" Reentrant="false" ReadOnly="false" Precondition="" Postcondition="" BodyFontSize="0">
```


Figure 3.21: Computer Representation of Work Station Level Integration

Integrated Design Cost Algorithm: - estimating the cost of integrated design is done at the workstation level for simplicity, however, this can also be done at the design level for more detailed cost estimation activities. Input parameters for the cost estimation of the integrated Process and Resources are:

Process and Resource Integration Cost Inputs

R_{Rm} = Resource Rate for machines

R_{Rh} = Resource Rate for human operators

T_{WCT} = Workstation Cycle Time

N_W = Number of Workstations

C_{PR} = Process and Resource Integration Cost

The cost estimation function for calculating the Process and Resource Integration Cost is given as:

$$C_{PR,l} = \sum_{l=1}^{n} (R_{R,l} \times T_{WCT,l} \times N_{W,l}) \tag{3.19}$$

Where l = *number of workstations = 1, 2, 3…, n*

$$R_{Rm} = \frac{Purchase\ Price - Depreciation}{Predicted\ Annual\ Hours\ of\ Operation} \tag{3.20}$$

$$R_{Rh} = \frac{Annual\ Salary\ of\ human\ Resource}{Expected\ Annual\ Working\ Hours} \tag{3.21}$$

(3) Process and Resource Integration Output

The outputs for Process and Resource Integration are:

- workstations and operations integration diagram;
- workstations and operations integration script and
- an Integrated Design Cost Equations at workstations level.

3.2.4.5 P-P-R Design Cost Calculator

An integrated cost model is developed as shown in Figure 3.22. the model brings together the product, process and resource design cost algorithms in a more graphical display in an easy to use interface. Each cost model contains fields for cost data inputs and then show the total design cost. The models are then integrated in such a way that that changes made in one cost model automatically reflect in the total cost as shown in Figure 3.22. Microsoft Excel is used for developing the cost calculator in this methodology as its output easily plugs in and interfaces with most major tools such as PLM and aPriori to drive other platforms.

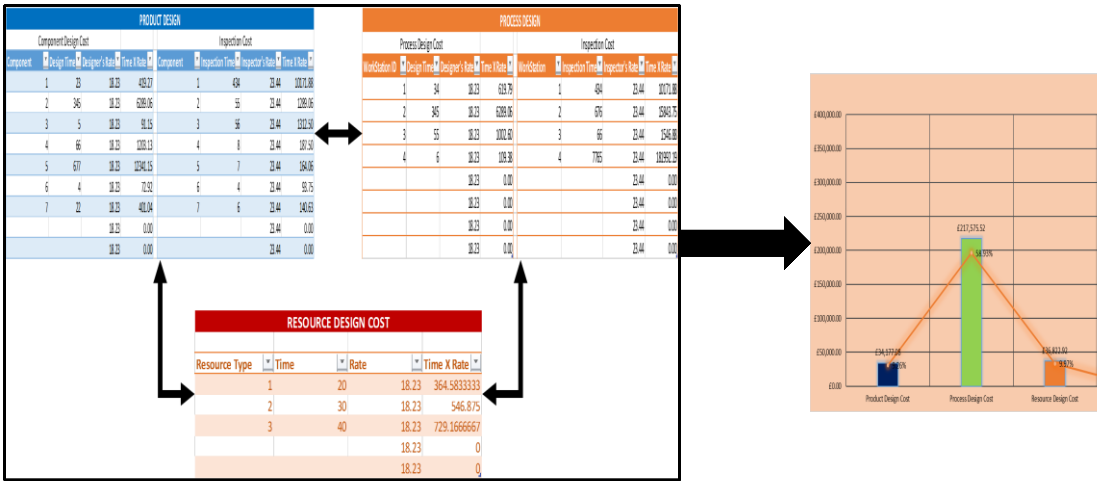


Figure 3.22: PPR Design Cost Calculator

This calculator serves as a cost summary for introducing cost as a common key performance indicator (KPI) between product designers and process designers where component design changes can be translated into cost in a single display.

3.3 A Technique for Extending Cost Modeller Capabilities to Include A New Process For Cost Assessment

This section looks at current challenges in cost estimation tools. State of the art techniques are identified as well as the research gaps in current approaches. A proposed approach has been introduced that attempts to address some of the identified gaps in research. The proposed approach is verified and validated in Chapter 4 by implementing an emerging welding technology called Remote laser Welding process into aPriori's workbench.

3.3.1 Knowledge based models in support of cost engineering

Research in knowledge related models in support of cost engineering has proven very useful in recent years (Agyapong-Kodua, Asare, et al., 2014; Tammineni et al., 2009). Knowledge-based cost models arose as a result of the drive to support engineering design activities with existing manufacturing, materials and cost databases (Tammineni et al., 2009). This approach has proven useful because cost is mainly knowledge intensive and rests on tacit knowledge of many discipline holders (Rush & Roy, 2001a, 2001c; E. Shehab & Abdalla, 2002). Agyapong-Kodua (Agyapong-Kodua et al., 2011) further explained that the set of relevant knowledge required by a cost engineer can be represented by Figure 3.23.

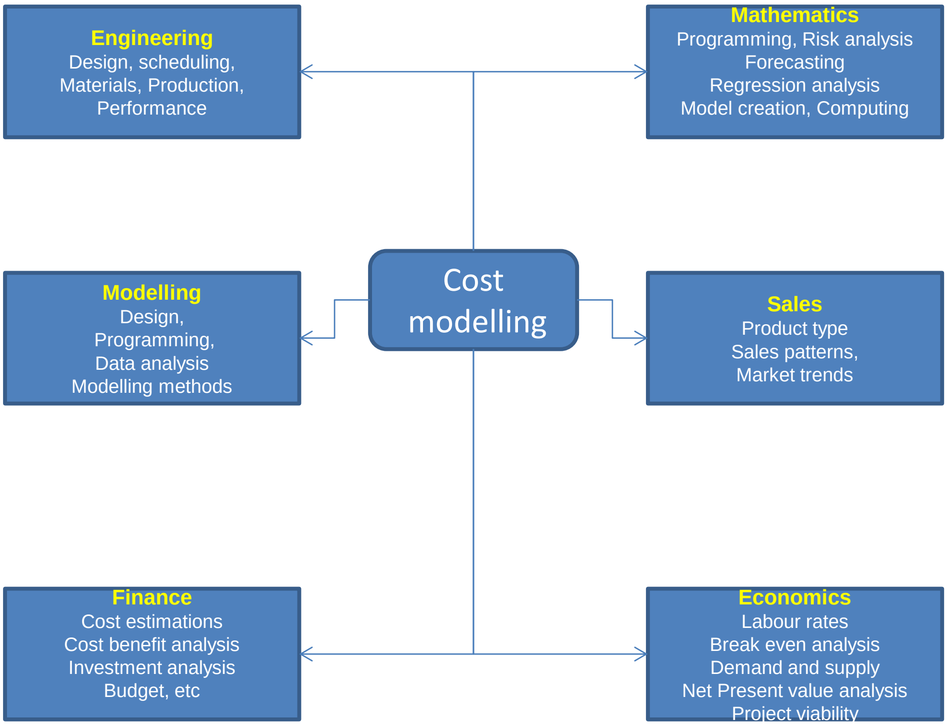


Figure 3.23: Relevant knowledge sets for a cost modeller (Agyapong-Kodua et al., 2011)

As a result cost modelling requires a knowledge-based system for dynamic capturing and representation of cost information for product cost estimation (Tammineni et al., 2009). A number of researchers have already deployed knowledge modelling techniques in the creation of cost estimation systems. Typical examples relate to Design Analysis Tool for Unit Cost Modelling (DATUM) (Scanlan et al., 2002) and aPriori (Agyapong-Kodua et al., 2014). Despite the success reported by these tools, Scanlan *et al.* (2002) conducted a study on knowledge-based cost modelling in aerospace designs and identified that there is the lack of generally accepted methods for dealing with uncertainty. Cheung *et al.* (2009) proposed a value-driven knowledge-based design model for use in modelling component costs in the aerospace industry.

They defined ‘value-driven design’ as the concept of concurrent engineering in which designers may utilise value models in determining the value of product designs being a single objective function. In their case, the cost model consisted of design attributes and a cost model database, both of which feed the Vanguard Studio software and ExtendSim. This model was applied to the development of an aero-engine fan blade at Rolls Royce (Cheung *et al.* (2009). Other supportive work to associate cost information with product design for use in the aerospace industry was provided by Tammineni (Tammineni et al., 2009). The research led to the development of a tool which is able to provide incremental cost fluctuations in response to changes in component geometry. This research achieved very useful outcomes but limited to the aerospace industry and also users have limited chance of interacting with the manufacturing systems model which is behind the cost engine. This is because the process modelling technique and cost estimation approach used were dedicated to the aerospace industry only based on the study of modelling and estimation techniques within the industry. This, therefore, makes it difficult to be applied in other engineering business domains. A similar knowledge based model for modelling the cost of designing composite wing structures in aircrafts was provided by Verhagen (Verhagen et al., 2010). Jin (Jin, Curran, Burke, & Welch, 2011) provided a very useful integration method for automated recurring cost prediction by employing digital manufacturing technology. The study developed a prototype tool for integrating assembly time cost and parts manufacturing costs, however, the authors focused on manufacturing cost rather than estimating the total cost to include investment, overheads, etc.

3.3.4 Proposed Research Technique

The proposed approach requires the modelling of product, process and resource that meets the cost estimation tool’s requirements.

3.3.4.1 Product-Process-Resource Models Execution Technique

The logic behind this is that models created meet the basic requirements of commercial cost modeller’s inputs that supports process extension of cost modellers. Executing

PPR models enables the user to view and modify Product, Process and Resource parameters to generate production process cost values. However, the execution process into a cost modeller requires knowledge of the cost modeller's input requirements. As an example, the data structure of a generic Cost Modeller requirements is shown in Figure 3.23. Mainly, a cost modeller's requirements consist of product requirement, Process requirement and resource requirement which has to be satisfied in order to generate cost results. In the Input Requirememnts in the Figure, an identification (ID) of Product, Process and Resource requirements in the form of names or numbers are assigned for easy identification. A description of the requirements are further expressed in a 'text' format to help understand the content of the requirements. Other information such as the source of data, verifying method, risks identified, and the status of the input requirements can all be computed at this stage.

These are satisfied mainly through product, process and resource models created based on the Product CAD Geometry. However, in many cases, most cost modellers have generic cost algorithms that integrate product, process and resources to generate cost estimates. Again, this methodology is focusing on how to create a new process in an existing cost modeller that is not capable of realising an innovative product.

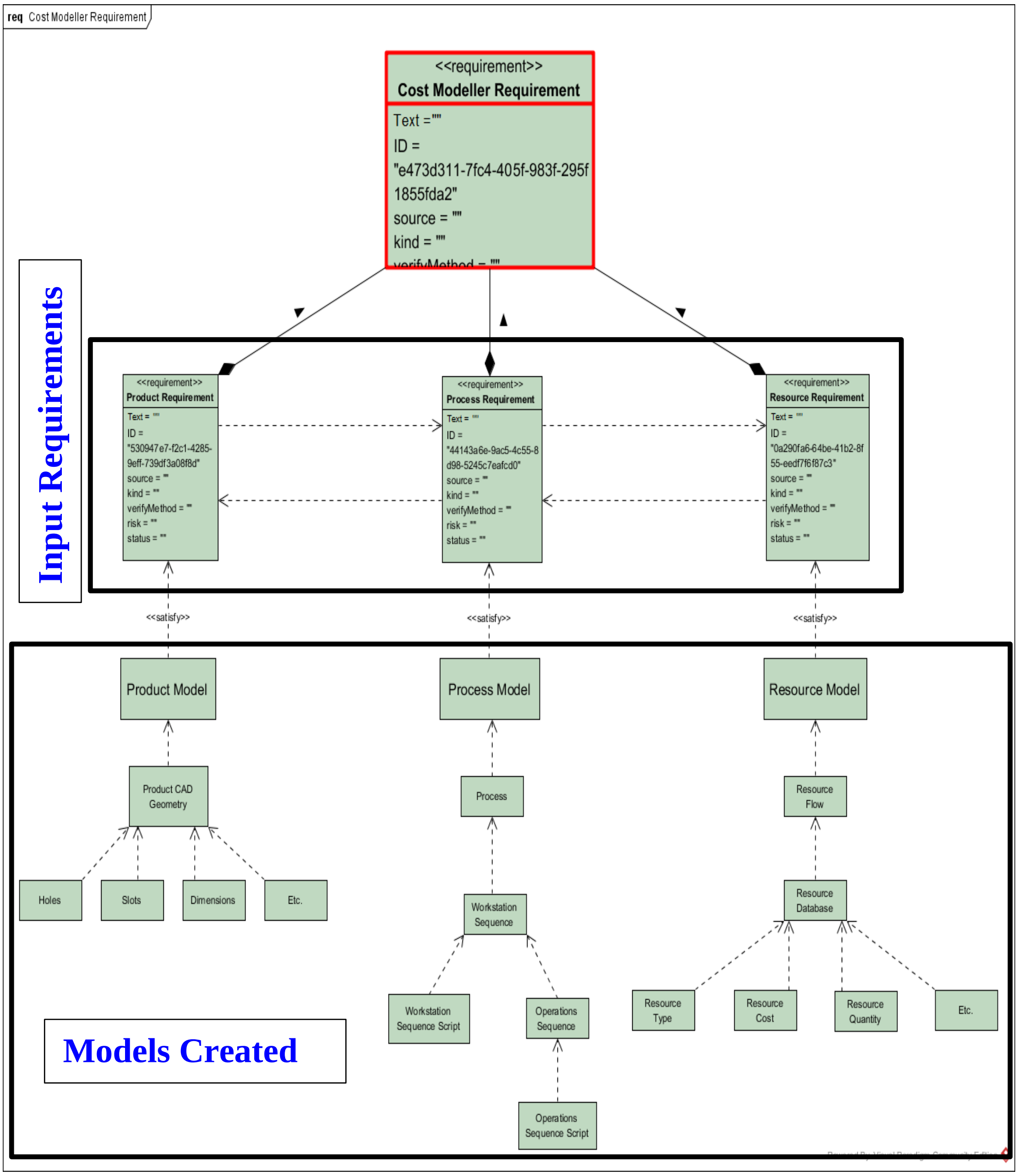


Figure 3.23: A Typical Cost Modeller Requirement Structure

Therefore, PPR Cost Estimation Framework suggests an initial listing of all inputs necessary for a cost modeller to function and then develop models that can satisfy those requirements. As the methodology considers estimating cost during early design stage where less data is available, knowing what data is available and the type of cost that can be calculated is an important aspect of the methodology. Most engineering cost estimation tools are only capable of calculation cost when all cost data are available and also based on the experience of the cost estimator. The aim of this approach is to develop a framework such that, cost data are generated and captured as they become available to generate better cost estimates to support engineering decision making. The following sections will detail how PPR Cost Estimation suggests the execution of modelled data into inputs required by cost modeller.

3.3.4.2 Cost Modeller Input Requirements

As cost modellers differ in various ways, models to be executed must satisfy the input requirements of the cost modeller to support the generation of cost values. An example of matching cost modeller requirements to developed models is shown in Figure 3.24, where the elements within the cost modeller are satisfied by the input models.

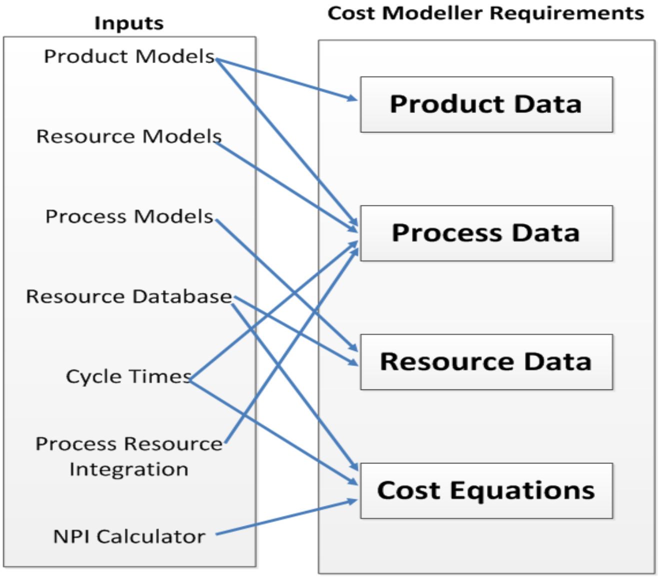


Figure 3. 24: A Sample Cost Modeller Requirements and Input

Because most cost modellers have inbuilt processes, they assume that resources are readily available to be consumed. In some cases, where there is an innovative product that requires specific processes, a cost modeller's inbuilt processes and resources are not able to generate cost values. In that case, PPR Cost Estimation Framework then becomes a good guide for expanding the capabilities of an existing cost modeller.

3.3.4.3 Product Data Requirement

Most cost modellers that have the capability of importing 3D CAD of a product often has inbuilt product feature recognition technology. This technology as discussed in

Chapter 2 has the ability to identify and retrieve details of product CAD model features that are already stored in the CAD model such as:

- Part Name or Number
- Mass (kg)
- Length (mm)
- Width (mm)
- Thickness (mm)
- Surface Area (mm^2)
- Volume (mm^3)
- Etc.

The features identification and recognition enables the cost modeller to select the most suitable processes and resources that are capable of realizing the product in order to generate cost values based on the product's features.

This proposed methodology requires that; (1) cost modeller allows users to import 3D CAD models (2) the cost modellers has product feature identifying, recognising and retrieving technology and (3) product CAD model is given as an input.

3.3.4.4 Process Data Requirement

In a more general sense, most cost modellers automatically show process options and the user selects that most suitable based on cost, process cycle time or other factors. However, to expand the capabilities of cost modellers to include new processes, the input requirements are:

Process Flow Diagrams: this is a graphical representation of how the process is expected to look like. This could be done at the workstation level for an overview of the process and also, at the operation level for a more detailed view of the workstations.

Scripting - codes are written to represent a manufacturing process and all its operations for realising a new product. This can be challenging as the code must be correct for the model to function properly as well as the ability of engineers to write the appropriate code. Usually, the scripting or codes are written in C, C+, C++, MATLAB, Java or other scripting languages. Examples of the scripts generated for Processes and operations using SysML are demonstrated in Figures 3.10, 3.16 and 3.19.

3.3.4.5 Resource Data Requirement

Expanding the capabilities of a cost modeller also requires building resource models and database that can be integrated with the manufacturing processes and to be consumed at various stages of the manufacturing process. The resource database has to be modelled in order to meet the requirements of the cost modeller. Most tools have a resource data structure of which the resource models and database has to be developed in the same data structure. Updating resource database in a cost modeller may be done in one of two ways:

Manually – where the user inputs the list of resources, their capabilities and their cost information.

Import – resource data may be created in a spreadsheet in a format that is compatible with the cost modeller's data structure.

PPR Cost Estimation Framework satisfies these requirements through its Resource Model and Resource Database.

3.3.4.6 Cost Equations Requirement

The cost equations are cost functions for estimating the cost of realizing a product. Most cost modellers directly use accounting techniques for calculating the cost processes. To create a cost function in a cost estimation tool may require access to its cost library. Cost functions are written in scripts, linking Product, Process and Resources cost information. The Cost Equations aim at calculating the unit cost or batch cost of a product based on its features displayed to users via a Graphical User Interface (GUI). IF_THEN statements are normally used in most scripting languages to create the cost equations in cost estimation tools. The equations have to be expressed in the format that can be understood by the cost modeller.

The PPR Cost Equations satisfies most cost equations although the presentations in other tools may vary, the concept and structures are usually the same for most estimation tools. An example of a cost equation using IF_THEN statements is:

> *GetMachineOverheadRate = { fail(msg(FLT, 'machine_overheadRate:Bad machine data. overheadRate is null. machine=', machine_name, FRT)) if (machine.workCenterOverheadRate == null)*
>
> *fail(msg(FLT, 'machine_overheadRate:Bad machine data. overheadRate must be >= 0. machine=', machine_name, FRT)) if (machine.workCenterOverheadRate < 0)*
>
> *machine.workCenterOverheadRate * plant.overheadRateAdjustmentFactor otherwise }*

Nevertheless, not all cost estimators allow users to implement new cost equations or allow new algorithms to be created by the user but rather, it remains a "black box" to users.

Once all models, database and equations are implemented and functioning well, the next stage will be to access the cost of the CAD Product, Process and Resources. However, where the implementation does not function well as expected, '*error messages*' pop up to help identify the cause. This helps in tracing back to the Data Modelling stage to make the necessary adjustments. A cross checking of cost results can be done by comparing the results obtained from the cost model with result obtained from the tradition cost estimation techniques using traditional cost accounting principles. This serves as a sanity check to validate the developed model.

3.3.4.7 Production Process Cost Equations

The production cost equation contains algorithms that are capable of calculating the cost of using a particular process. This is the sum of Fixed Cost and Variable Cost. These equations are programmed in the cost modeller using the appropriate programming language. The major components of the cost equations are mainly the fixed and variable cost equations.

a) ***Fixed Cost Equations*** - these are the costs that remain the same even with the increase in the level of production. This cost also refers to the expense incurred because of running a system. Fixed cost in the methodology is limited to costs of purchasing machines, tools and other engineering related costs such as Robot Cost, Hard Tooling Cost, Fixture Cost, Programming Cost, Additional Amortised investment, etc.

Fixed Cost Input

$C_F = Fixed\ Cost$

$C_m = Purchasing\ Cost\ of\ Machines$

$C_{in} = Cost\ of\ Machine\ Installation$

$$C_F = \sum(C_m + C_{in}) \tag{3.29}$$

$$C_{m,t} = \sum_{t=1}^{n}(C_{m,1} + C_{m,2} + C_{m,3} + \cdots \ldots\ldots C_{m,t}) \tag{3.30}$$

Where

$t \geq 1$; Number of machines

b) Variable Cost Equations - Unlike fixed costs, variable costs changes in direct proportion to the number of outputs which also increases when activities increase and decrease when activities decrease. PPR Cost Estimation classifies Variable Costs as material, labour and energy costs usually consumed during a manufacturing process.

Variable Cost Inputs

$$C_{var} = Variable\ Cost$$

$$C_{var} = \sum(C_{res} + C_{ene} + C_{mat}) \tag{3.31}$$

Where;

$\boldsymbol{C_{res} = Resource\ Cost}$

$$Resource\ Cost = Labour\ Cost + Machine\ Cost \tag{3.32}$$

$$Labour\ Cost = Labour\ Rate \times Labour\ Time \tag{3.33}$$

$$Machine\ Cost = Machine\ Rate \times Machine\ Time \tag{3.34}$$

$\boldsymbol{C_{ene} = Energy\ Cost}$

$$\boldsymbol{Energy\ Cost}\ (\boldsymbol{C_{ene}}) = R_{ene} \times T_{ene} \tag{3.35}$$

Where;

$R_{ene} = Energy\ Rate$

$T_{ene} = Energy\ Utilisation\ Time$

$C_{mat} = Material\ Cost$

Where;

$$R_{ene} = Energy\ Rate$$

$$T_{ene} = Energy\ Utilisation\ Time$$

$$\boldsymbol{Material\ Cost}\ (\boldsymbol{C_{mat}}) = (cost\ per\ kg\ X\ Part\ Mass)\ /\ utilization \tag{3.36}$$

$$Rough\ Mass = \frac{Finish\ Mass}{Utilization} \tag{3.37}$$

$$Scrap\ Mass = \frac{rough\ Mass}{finish\ Mass} \tag{3.38}$$

$$Finish\ Mass = Part\ volume * material\ density \tag{3.39}$$

3.3.4.8 Installation Cost Equations

The cost of installation is the cost of physically building the manufacturing system. This is the sum of mechanical installation cost, electrical installation cost and IT installation cost. Although other costs may be associated with installation, this research is limiting installation cost to the above. The following equations are generated to calculate the cost of installing a manufacturing system.

$$C_{in} = Installation\ Cost$$

$$C_{in} = \sum (C_{mec} + C_{elect} + C_{it}) \tag{3.22}$$

Where;

$$C_{mec} = Cost\ of\ Mechanical\ Installation$$

$$C_{mec} = R_{mec} \times T_{mec} \tag{3.23}$$

Where;

$$T_{mec} = Mechanical\ Installation\ Time$$

$$R_{mec} = Installer\ rate$$

$$R_{mec} = \frac{Annual\ Installer\ Salary}{Predicted\ Annual\ Hours} \tag{3.24}$$

$$C_{elect} = Cost\ of\ Electrical\ Installation$$

$$C_{elect} = R_{elect} \times T_{elect} \tag{3.25}$$

Where;

$$T_{elect} = electrical\ Instalation\ Time$$

$$R_{elect} = \frac{Annual\ Installer\ Salary}{Predicted\ Annual\ Hours} \tag{3.26}$$

$$C_{it} = Cost\ of\ IT\ Installation$$

$$C_{it} = R_{it} \times T_{it} \tag{3.27}$$

Where;

$$T_{it} = IT\ Installation\ Time$$

$$R_{it} = \frac{Annual\ Installer\ Salary}{Predicted\ Annual\ Hours} \tag{3.28}$$

3.3.5 Cost Assessment

The next stage of the methodology is the assessment of the product cost based on a selected process. A product CAD model containing geometric features can be imported into the cost modeller's environment to assess the cost of realising the product features by selecting appropriate processes and resources. The assessment of cost is done to various process options selection based on cost and process viability to support engineering decision making. Cost may be assessed for:

- Piece part cost

- Bulk cost
- Material type cost
- Batch size
- Total variable cost
- Fully burdened cost
- Number of weld stitches (for welding process) cost.

3.4 A Technique for Integrating P-P-R-Production Cost Values to Support Engineering Decisions.

An integrated cost calculator is developed by integrating the PPR design cost models together with the installation and production cost. From the literature review in Chapter 2, it was observed that there are models and techniques developed for estimating product cost. It was also observed that, current cost estimation tools used for estimating the cost of engineering product only considers the production cost of the product. From literature, the design cost of product, process and resources are treated as an overhead cost which is mainly calculated by cost accountants using techniques such as activity based costing or expert judgement.

An integrated P-P-R-Production Cost Estimation Technique is proposed which integrates product design cost, process design cost, resource design cost, installation cost and production cost models that graphically displays the total cost of a product. The integrated cost model is interactive in nature such that changes made to any cost component automatically reflect in the total cost graphically in the form of a cost summary as shown in Figure 3.25. The cost calculator may also serve as a cost dashboard for people working and overseeing multiple product lines to ensure that cost values are always visible graphically and can immediately identify areas which needs attention.

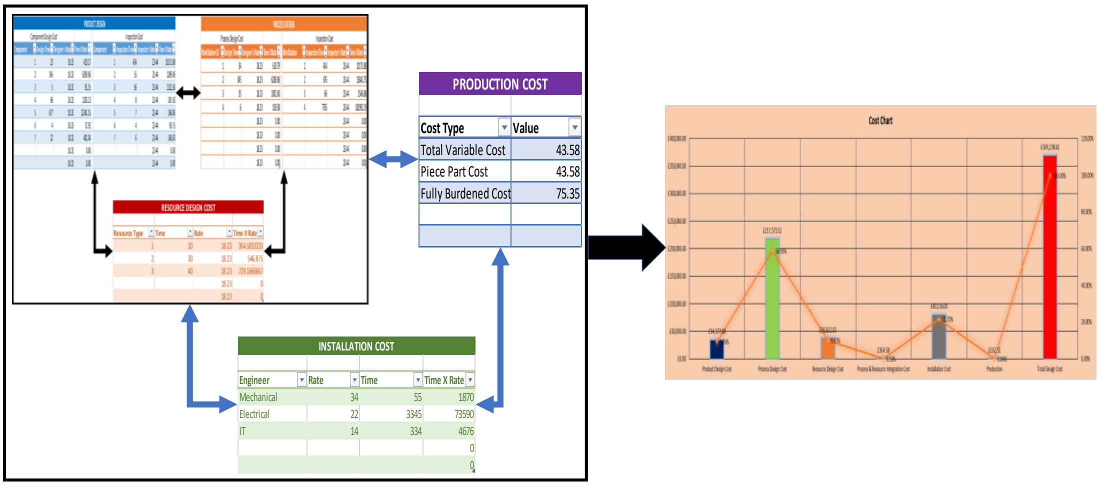


Figure 3.25: Integrated Cost Calculator

3.4.1 Models Execution into Cost Modeller to Extend Its Capabilities To Include Remote Laser Welding Process For Cost Assessment

The cost modeller used for RLW process cost estimation is aPriori, a commercial of the shelf software used for product cost analysis. The software uses feature identification and recognition techniques for identifying CAD model geometric features. These features are the geometric cost drivers (GCDs) and they are assessed by the CAD model complexity, material properties, size, weight, tolerances and surface finish. The software in addition factors in non-geometric cost drivers such as manufacturing process selection, material cost, production volume and the facility where the manufacturing will be carried out. The structure behind aPriori is to capture a particular factory's manufacturing capability as a virtual production environment (VPE). These contain information on machines and process that are capable of realising the product. These then help in generating a product cost assessment to support engineering decisions.

The system architecture for aPriori is shown in Figure 3.26. aPriori has a Cost Engine which contains the Virtual Production Environment (VPE) and the Cost Model Workbench (CMWB). The VPE and the CMWB interact with each other at lower levels within the software which makes the integration of Product, Process and Resource data possible through scripting and coding.

However, some of the limitations of aPriori are that it contains predefined processes and resources in its database, which assumes resources are 100% available at each given time. The software also assumes an ideal manufacturing processes with appropriate resources which in practice may not exist. Hence, where there are improvement or changes to a process or resource information or type, aPriori is incapable of realizing those changes to reflex reality.

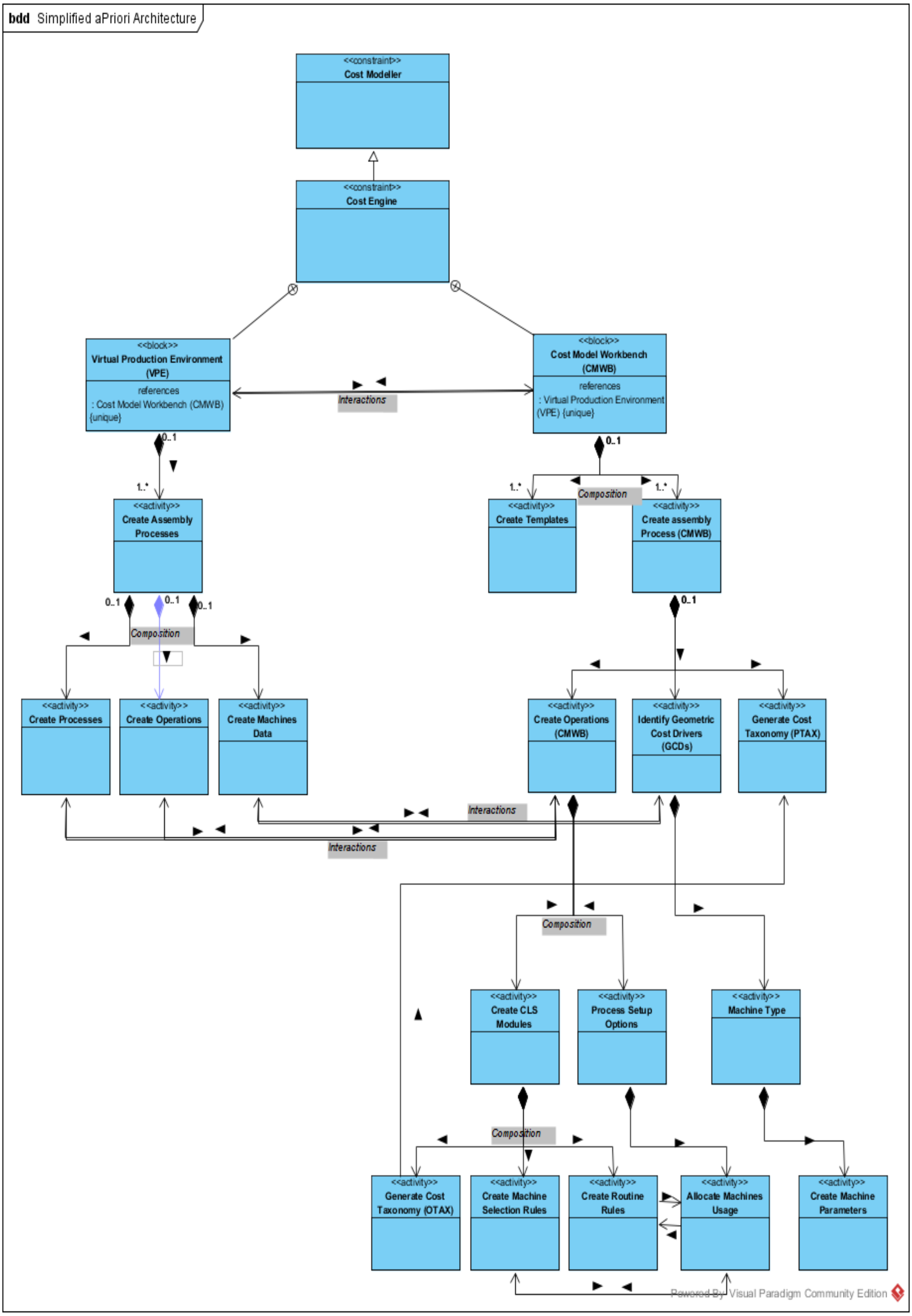


Figure 3.26: Cost Modeller Architecture

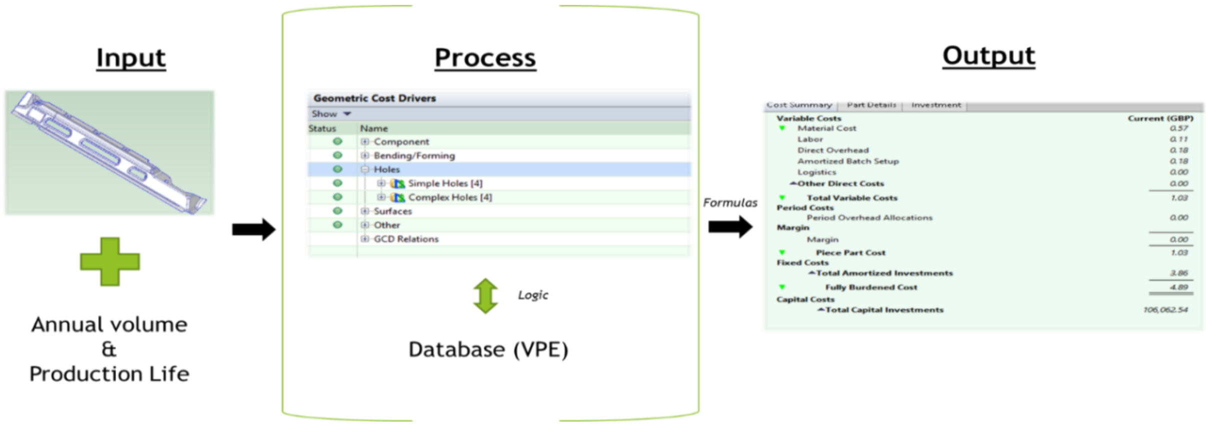


Figure 3. 1: Working Principle of Cost Modeller

Generally, using the cost modeller requires importation of a CAD model (Figure 4.18), the cost modeller then extracts features from CAD models (e.g. holes, bending, cutting, etc.) and generates a list of Geometric Cost Driver (GCD). It then fetches required data from the VPE guided by logic/ rules and formulas behind the VPE, it then determines various manufacturing processes with their times and costs.

The cost modeller is, however, only able to cost the entire CAD model based on the geometric features identifies but it is not possible for it to cost individual features on a particular CAD. Also, process selection within the software does not include that of remote laser welding with its resource types, resource values and process cycle times.

To make RLW as a process option within the cost modeller requires a rigorous process of creating a dedicated database containing process model, process rules, resource model, cycle times, process logic, computation of cost equation to reflect the process and finally resource database creation containing their cost values and utilization rates. An overview of the feature cost estimating model is shown in Figure 3.27, highlighting the new inputs in the cost modeller.

3.4.2 Generation P-P-R Cost Report

Reports are then generated based on cost by changing the various product, process and resource parameters to see its implications on cost. Cost report could be based on an individual component of the product or on a product as a whole. For most cost modellers, cost reports mainly give information on:

- Piece Part Cost Report
- Bulk Cost Report
- Assembly Cost Report
- Cost Comparison Report
- All Components Cost Report

Reports on cost information may also be generated by the manufacturing process used to realize the product which may also include:

- Manufacturing cost
- Assembly process cost
- Workstation cost
- Operational cost

However, a comprehensive cost report may contain a breakdown of:

- Variable costs
- Margin costs
- Fixed costs and
- Capital cost reports.

Another report on for example Resource utilization involving human operators, machines and tooling may also include cost information such as:

- Lifetime tooling cost
- Labour cost
- Labour rate
- Labour time
- Machine cost
- Machine hourly rate

Although these lists of information are not exhaustive, these are more generic information contained in most cost estimation reports that are able to support engineering decision making. These may be represented in graphs, charts, in tables or a combination of all. An example of a report generated using SEER-DFM cost estimation tool is shown in Figure 3.26. This is an alternative cost modeller that that could be integrated with PPR methodology as the tool has the capability of recognising product features but not able to generate early design stage costs for product, process and resource design.

Figure 3. 26: Sample of Report form SEER-DFM (http://galorath.com, 2016)

3.5 Summary

Based on the review of the literature, research gaps were identified and an integrated methodology coined PPR Cost Estimation Framework was proposed for addressing the gaps identified. The proposed framework consists of three components; A Product-Process-Resource Modelling Technique for Capturing Engineering Knowledge and Cost Values (involves the modelling of P-P-R data which includes representation, illustration and design cost calculations), A Technique for Extending Cost Modeler Capabilities to Include a New Process For Cost Assessment (implementation of P-P-R models into cost modeller to extend its capabilities) and An Integrated Product-Process-Resource-Production Cost Estimation Technique (an integrates cost model that displays a cost summary). The proposed PPR Cost Estimation Framework bridges the gap between design and manufacturing by introducing cost as a key performance indicator.

CHAPTER 4

CASE APPLICATION

Early design stage cost estimation of a new product or a new process is very crucial, particularly in a competitive market. Information for cost estimation for some products may be readily available particularly when the product to be manufactured is similar to an existing one or is a modification of current models. This is also true for new processes. In the case of a cutting-edge technology, it becomes a challenge to make a good cost estimate with limited or no product, process and resource data. Much research has been done in the past decade in the area of engineering evaluation for new product introduction (NPI) [(Fuchs et al. 2010), (Arakji & K.R. 2007), (Talay et al. 2014), (Kotler & Armstrong 2012), (Sandmeier et al. 2010), (Amue & Adiele 2012)] . However, there is yet the need to further investigate how to estimate the cost of a new product at the early design stage of a new product introduction or development that enables engineers to predict the cost effect on engineering changes made on a product. PPR Cost Estimation Framework, therefore, is an approach to supporting engineering decision making the process by generating viable options in implementing a new key enabling technology (KET) into an existing production system for product cost assessment.

The PPR Cost Estimation Framework shows how information is collected and transformed into models that become inputs to a commercial of the shelf (COTS) cost estimator for cost assessment. The methodology required the extension of the cost estimator (software) and the approach to this is demonstrated in this case application.

4.1 Remote Laser Welding (RLW) Navigator Project

Remote Laser Welding (RLW) Navigator is a three year, European Commission sponsored project under the ICT-Factories of the Future programme which began in January 2012 to June 2015 with fourteen industrial and academic project partners. The goal of the research project was to develop an engineering platform for an emerging joining technology (Remote Laser Welding) specifically for the automotive industry that will enable the exploitation of this technology and ultimately support other joining processes. The RLW Navigator project developed new software-based tools for systematic and rapid development and deployment of RLW technology into automotive body production systems. The tools developed provides the missing capabilities to simulate products and processes from design to production, hence facilitating '*right-first-time'* capabilities of production systems. The developed tools include:

- production system-level configurator with assembly layout and process estimator;
- a workstation planning and RLW off-line programming (OLP);
- a process optimiser with part variation modeller, fixture layout analyser and optimiser, and laser parameters optimiser;
- process control with weld quality performance evaluator and
- an eco-advisor (http://www.rlwnavigator.eu/).

The developed tools were successfully applied in a practical experiment in a test cell at the Warwick Manufacturing Group (WMG) on a JLR's Range Rover Evoque SUV door. The results revealed some benefits for the RLW technology:

✓ 60% less shop floor space requirements
✓ The use of 5 industrial robots instead of 14 for a comparable RSW process
✓ Shorter cycle time compared with current RSW door welding technology.
✓ provides opportunities for enhanced product design by reducing and elimination current processes (http://www.automotivemanufacturingsolutions.com/)

These results were obtained based on simulation using although there were no baseline for the technology, however, in comparison to spot welding, better cost benefits may be realised. This is because RLW process is faster due to the use of better resources such as fast robots with laser technology and the use of customised fixtures.

The Remote Laser Welding (RLW) Navigator project is used as a case study to validate the PPR Cost Estimation Framework. The research aims at estimating and analysing manufacturing cost at the design stage using RLW technology on a car. For a new process, such as RLW, it is not entirely clear from the outset what these might be, how they might interact and influence the overall cost. Validating the genericity of the methodology, RLW data will be modelled and to follow all necessary stages and steps as described in Chapter 3 (Figure 3.1).

4.2 PPR Cost Estimation Framework Implementation for Remote Laser Welding Process

The RLW Navigator Project was intended to provide a single software toolkit to facilitate the process planning, design, implementation and optimisation in the application of Remote Laser Welding technology in Body In White sheet metal joining domain. The toolkit was designed to meet design and engineering key performance indicators (KPI) such as:

- Improved joint quality
- Facilitate parameter selection based on process performance
- Facilitate Statistical Process Control and root cause of joint failure
- Capability for in-line closed loop process control and adjustment
- Estimates energy used for welding
- Animates given path to show robot movements
- Calculates energy for robot movement calibrated by experiments etc.

However, it is obvious that the functionalities of the toolkit was the priori focus and cost was not considered as a KPI amongst. Since it is a novel technology with no prior costing information, using traditional cost accounting techniques was not an appropriate cost estimation approach. PPR cost estimation framework was therefore introduced to ensure that cost was introduced as a KPI for the research project amongst the other design and engineering KPIs stated above.

Figure 4.1 shows the framework of the methodology as applied to the RLW Navigator Project to validate the methodology.

The implementation will follow all the steps described in chapter 3 of this thesis. 3D CAD model of a car door consisting of 8 components are given as an input to the methodology. The Remote Laser Welding Product-Process-Resource (PPR) models are then created for graphical illustration and computer representation. A cost calculation is developed as a cost model, using standard cost accounting algorithms. The cost model is used for calculating the product process and resource design cost of the RLW process. The decision gate checks and validates the models designed as well as approval of design cost values. This supports the capturing of engineering knowledge and cost values.

The models developed for the RLW process and resources are implemented into a cost modeller (aPriori) to extend its capabilities. The computer representation of aspect of the models created earlier makes it easier to execute the models based on the software's requirements. This therefore demonstrates the technique for extending cost modeller capabilities to include a new process for cost assessment.

Also, the integrated cost estimation dashboard as shown in Figure 4.1 combines the PPR design cost and the production cost values to support engineering cost decisions. This integrated dashboard displays instant cost values where there are changes to input parameters. This there becomes a useful tool for running scenarios.

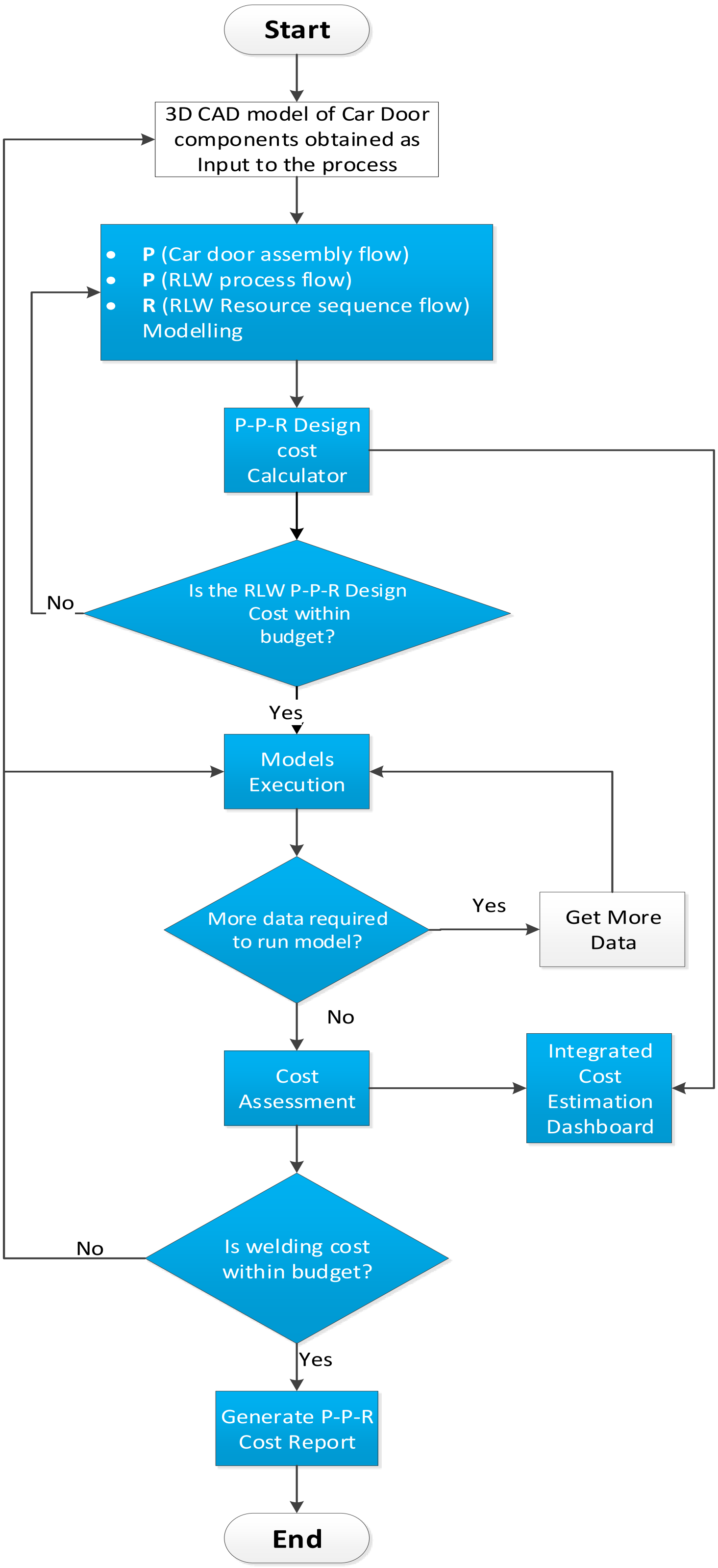


Figure 4. 2: RLW Application of PPR Cost Estimation Framework

4.3 3D CAD of Car Door Model

A 3D product CAD models of a car door is used to verify and validate the PPR Cost Estimation Framework. The CAD models is used as an input to the framework and it is made up of individual product components of an assembled car door. The car door consists of 8 different components received from product designers. The CAD models contain geometric data such as part dimensions, tolerance, design features, weight, etc.

The component list of the door assembly are:

- Halo Sub-Assembly
- Hinge Re-inforcement
- Latch Re-inforcement
- Halo
- Window Channel
- Belt Re-inforcement
- 2 Hinge Plates
- Door Inner Panel

These components are shown in Figure 4.2 as 3D CAD models.

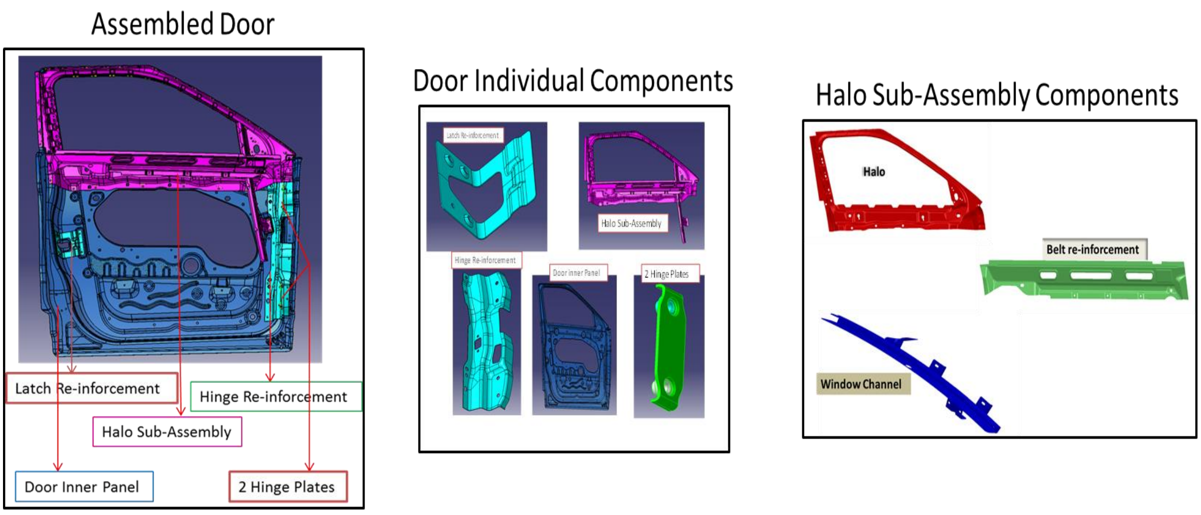


Figure 4. 3: 3D Door Components

Remote Laser Welding (RLW) technology requires welding from a thinner material onto a thicker material. Due to the material thickness of the two hinge plates and latch re-inforcement, they are introduced into the welding fixture first, followed by the door inner panel, then the halo sub-assembly and finally, the hinge-reinforcement.

4.4 RLW PPR Data Modelling

Modelling data as describes in Chapter 3 includes the following three [*Illustration* (model-based), *Computer Representation* (script) and *Cost Estimation Algorithms]* modules.

Data is modelled for:

- RLW Product Models
- RLW Process Models
- RLW Resource Models
- RLW Process and Resource Integration.

4.4.1 RLW Product Model

The product model for RLW consists of inputs, modules and outputs.

4.4.1.1 RLW Product Model Inputs

There are two main inputs for modelling the RLW product;

1. Product CAD components (Halo Sub-Assembly, Hinge Re-inforcement, Latch Re-inforcement, Halo, Window Channel, Belt Re-inforcement, 2 Hinge Plates, Door Inner Panel).
2. Cost Estimation Formulae – these include the product design cost calculation; designer's rate and designer's working hours.

4.4.1.2 RLW Product Modules

(1) RLW Product Tree

The product tree shows the relationship between the door components to be welded using the RLW technology. As shown as a graphical representation in Figure 4.3, the product tree is useful to the process designer for setting up workstations for the process to realize the final assembly.

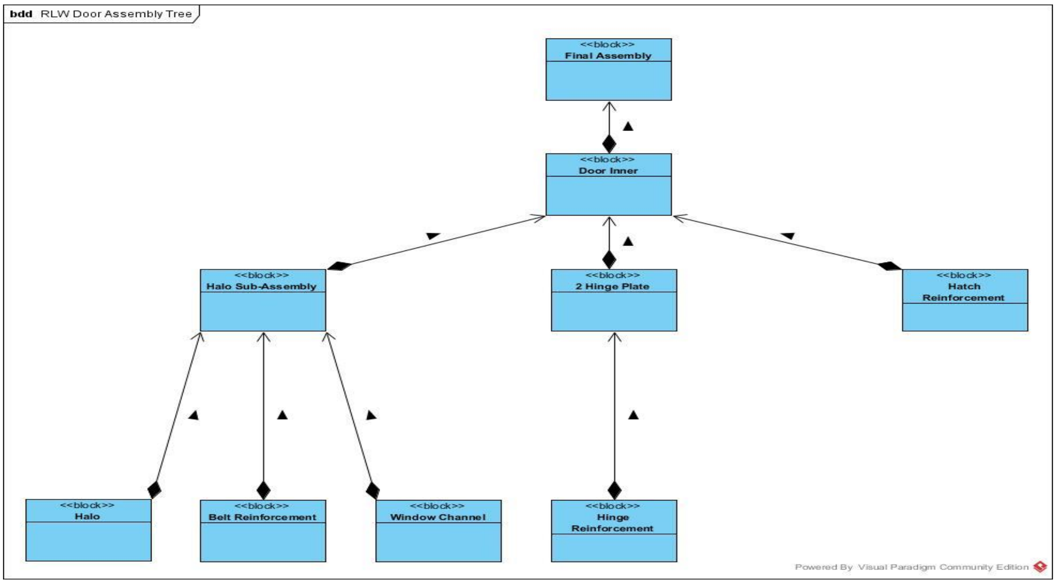


Figure 4. 4: RLW Door Assembly Tree

(2) RLW Computer Representation

A computer representation of the RLW product tree is generated as shown in Figure 3, where changes made in the script also changes the computer representation in the graphical user interface. The computer representation enables the cost modeller to identify product components based on component identification names as highlighted in Figure 4.4 for Halo sub-Assembly and Hatch reinforcement.

```
<Stereotype Name="block" Id="spnfcWqESzVmxyJt" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-08-15T17:23:51.403" PmCreateDateTime="2016-06-29T15:11:25.625"
  PmAuthor="asare_k" Documentation_plain="" Root="false" Leaf="false" IconPath="" BaseType="SysMLBlock" Abstract="false"/>
+ <ModelRelationshipContainer Name="relationships" Id="svQg8WqESzVmxyMZ" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2017-02-27T18:46:30.034" PmCreateDateTime="2016-08-
15T17:23:51.373" PmAuthor="asare_k" Documentation_plain="">
+ <SysMLBlock Name="Final Assembly " Id="bL6rAVqESzVmxwZX" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2017-02-27T18:46:30.032" PmCreateDateTime="2017-02-27T18:29:19.286"
PmAuthor="asare_k" Documentation_plain="" Root="false" Leaf="false" Abstract="false" Visibility="public" ConnectToCodeModel="1" Encapsulated="false" BusinessModel="false" BusinessKeyMutable="true"
Active="false">
+ <SysMLBlock Name="Door Inner" Id="7_trAVqESzVmxwZj" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2017-02-27T18:46:30.033" PmCreateDateTime="2017-02-27T18:30:50.103"
PmAuthor="asare_k" Documentation_plain="" Root="false" Leaf="false" Abstract="false" Visibility="public" ConnectToCodeModel="1" Encapsulated="false" BusinessModel="false" BusinessKeyMutable="true"
Active="false">
+ <SysMLBlock Name="Halo Sub-Assembly" Id="qYbrAVqESz mxwZ0" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2017-02-27T18:46:30.035" PmCreateDateTime="2017-02-27T18:31:23.285"
PmAuthor="asare_k f="false" Abstract="false" Visibility="public" ConnectToCodeModel="1" Encapsulated="false" BusinessModel="false" BusinessKeyMutable="true"
Active="false">
+ <SysMLBlock Name="Hinge Reinforcement" Id="pibrAVqESzVmxwZ9" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2017-02-27T18:46:30.035" PmCreateDateTime="2017-02-27T18:31:24.005"
PmAuthor="asare_k" Documentation_plain="" Root="false" Leaf="false" Abstract="false" Visibility="public" ConnectToCodeModel="1" Encapsulated="false" BusinessModel="false" BusinessKeyMutable="true"
Active="false">
+ <SysMLBlock Name="2 Hinge Plate" Id="zWbrAVqESzVmxwaG" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2017-02-27T18:46:30.033" PmCreateDateTime="2017-02-27T18:31:24.595"
PmAuthor="asare_k" Documentation_plain="" Root="false" Leaf="false" Abstract="false" Visibility="public" ConnectToCodeModel="1" Encapsulated="false" BusinessModel="false" BusinessKeyMutable="true"
Active="false">
+ <SysMLBlock Name="Hatch Reinforcement" Id="jJbrAVqESzVmxwaP" serIDLastNumericValue="0" QualityScore="-1" PmLastModified="2017-02-27T18:46:30.036" PmCreateDateTime="2017-02-27T18:31:25.233"
PmAuthor="asare_k bstract="false" Visibility="public" ConnectToCodeModel="1" Encapsulated="false" BusinessModel="false" BusinessKeyMutable="true"
Active="false">
+ <SysMLBlock Name="Halo" Id="OcnrAVqESzVmxwaY" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2017-02-27T18:46:30.036" PmCreateDateTime="2017-02-27T18:31:36.092"
PmAuthor="asare_k" Documentation_plain="" Root="false" Leaf="false" Abstract="false" Visibility="public" ConnectToCodeModel="1" Encapsulated="false" BusinessModel="false" BusinessKeyMutable="true"
Active="false">
+ <SysMLBlock Name="Belt Reinforcement" Id="PqnrAVqESzVmxwah" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2017-02-27T18:46:30.035" PmCreateDateTime="2017-02-27T18:31:36.572"
PmAuthor="asare_k" Documentation_plain="" Root="false" Leaf="false" Abstract="false" Visibility="public" ConnectToCodeModel="1" Encapsulated="false" BusinessModel="false" BusinessKeyMutable="true"
Active="false">
+ <SysMLBlock Name="Window Channel" Id="qDnrAVqESzVmxwaq" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2017-02-27T18:46:30.036" PmCreateDateTime="2017-02-27T18:31:38.261"
PmAuthor="asare_k" Documentation_plain="" Root="false" Leaf="false" Abstract="false" Visibility="public" ConnectToCodeModel="1" Encapsulated="false" BusinessModel="false" BusinessKeyMutable="true"
Active="false">
</Models>
+ <Diagrams>
</Project>
```


Figure 4. 5: RLW Computer Representation

Due to the feature recognition abilities of the cost modeller used, major features on the individual components such as holes, slot, bends, etc. can be extracted.

(3) ***RLW Product Design Cost Estimation Algorithm***

This calculates the design cost of individual components that make up the overall product. However, estimates are given for the time required by designers to design the various components. Using equations in Chapter 3 section 3.2.1.2, with given cost values such as designer's annual salary of £40000 and estimated weekly working hours of 40hr (based on 8hrs a day and 5 days a week), the cost of all components can be estimated. Again, expert's experience was required for the time required to design the individual components. Using equation (3.3), the designers' rate and inspectors' rate were calculated as:

$$R_D = \frac{£40000}{40hrsX48wks}$$

$$= £20.83/Hour$$

Therefore, we can say that designer's rate is approximately £21/hr.
It is required that an inspector approves the product designs. The inspector's annual salary for the RLW was set to £52000. Using equation (3.5), the inspector's rate is;

$$Design\ Inspector's\ Rate\ (R_{Di}) = \frac{£52000}{40hrsX48wks}$$

$$= £27/hr$$

Cost values obtained from the cost model were first compared with rates paid by job advertised online and on company websites and the results showed similar values. Secondly, the values were verified by two industrial experts from different companies who were partners of the RLW Navigator project confirmed that the cost values are true reflection of industrial rates.

4.4.1.3 RLW Product Components Cost Calculations

Having the inputs required for estimating the design cost of the various product components, equations (3.2) and (3.4) were used for calculating the design and inspection costs for individual components. Equation (3.7) is then used to calculate the subtotal cost of the product component. Hence, the total estimate for a component is the sum of design and inspection costs. The total product design cost is calculated using equation (3.1), where all the component costs are summed up. These equations are used to calculate the design cost estimate for the RLW Navigator door.

Halo

Inputs

It is assumed that it will take:

Design hours – 200

Designer Rate – £19.27/hr

Inspection hours of – 10

Inspector Rate - £26.04/hr

Number of Designers - 1

Having the inputs for Halo design, the design cost was calculated using equation (3.2).

$$C_{D,Halo} = 19.27 \times 200 \times 1$$
$$= £3854$$

Halo Design Inspection Cost was calculated using equation (3.4) as;

$$C_{D,Halo(INSP)} = 26.04 \times 10 \times 1$$
$$= £260.4$$

Therefore, Subtotal Halo Design Estimate

$$C_{Halo} = \sum (C_{D,Halo} + C_{D,Halo(INSP)})$$
$$= 3854 + 260.4$$
$$=£4114.4$$

Door Inner Panel

Inputs

Design hours – 340

Inspection hours of – 50

Number of Designers – 1

$$C_{D,DoorInnerPanel} = 19.27 \times 340 \times 1$$
$$C_{D,DoorInnerPanel} = £6551.8$$

Door Inner Panel Design Inspection Cost

$$C_{D,DoorInnerPanel(INSP)} = 26.04 \times 50 \times 1$$
$$C_{Di,DoorInnerPanel(INSP)} = £1302$$

Subtotal Door Inner Panel Design Cost was calculated using equation (3.7), where the total cost is given as:

$$C_{D,DoorInnerPanel} = \sum(£6551.8 + £1302)$$
$$=£7853.8$$

Door Fixture

Inputs

Design hours – 400

Designer Rate – £19.27/hr

Inspection hours of – 100

Inspector Rate - £26.04/hr

Number of Designers – 1

Door Fixture Design Cost

$$C_{D,DoorFixture} = 19.27 \times 400 \times 1$$
$$= £7708$$

Door Fixture Design Inspection Cost

$$C_{D,DoorFixture(INSP)} = 26.04 \times 100 \times 1$$
$$= £2604$$

Subtotal Door Fixture Design Estimate

$$C_{DoorFixture} = \sum £7708 + £2604$$
$$=£10,312$$

Latch Re-inforcement

Inputs

Design hours – 40

Designer Rate – £19.27/hr

Inspection hours of – 5

Inspector Rate - £26.04/hr

Number of Designers – 1

Latch Re-inforcement Design Cost

$$C_{D,\text{LatchRe-info}} = 19.27 \times 40 \times 1$$

$$= £770.8$$

Latch Re-inforcement Design Inspection Cost

$$C_{D,\text{LatchRe-info(INSP)}} = 26.04 \times 5 \times 1$$

$$= £130$$

Subtotal Latch Re-inforcement Design Cost

$$C_{Est,\text{LatchRe-info}} = \sum(C_{D,\text{LatchRe-info}} + C_{D,\text{LatchRe-info(INSP)}}) \tag{4.13}$$

$$C_{\text{LatchRe-info}} = \sum(£770.8 + £13002)$$

$$=£901$$

Hinge Re-inforcement

Inputs

Design hours – 70

Designer Rate – £19.27/hr

Inspection hours of – 10

Inspector Rate - £26.04/hr

Number of Designers – 1

Hinge Re-inforcement Design Cost

$$C_{D,\text{HingeRe-info}} = 19.27 \times 70 \times 1$$

$$= £1348.9$$

Hinge Re-inforcement Design Inspection Cost

$$C_{D,\text{HingeRe-info}} = 26.04 \times 10 \times 1$$

$$= £260.4$$

Subtotal Hinge Re-inforcement Design Cost was calculated using equation (3.7) as

$$C_{D,\text{HingeRe-info}} = \sum £1348.9 + £260.4$$

$$=£1609.3$$

Hinge Plate

Inputs

Design hours – 30

Designer Rate – £19.27/hr

Inspection hours of – 10

Inspector Rate - £26.04/hr

Number of Designers – 1

Hinge Plate Design Cost

$$C_{D,\mathrm{HingePlate}} = 19.27 \times 30 \times 1$$
$$= £578.1$$

Hinge Plate Design Inspection Cost

$$C_{D,\mathrm{HingePlate(INSP)}} = 26.04 \times 10 \times 1$$
$$= £260.4$$

Subtotal Hinge Plate Design Cost

$$C_{\mathrm{HingePlate}} = \sum(£578.1 + £260.4)$$
$$=£838.5$$

Belt Reinforcement

Inputs

Design hours – 30

Designer Rate – £19.27/hr

Inspection hours of – 10

Inspector Rate - £26.04/hr

Number of Designers – 1

Belt Reinforcement Design Cost

$$C_{D,BeltReinf} = 19.27 \times 30 \times 1$$
$$= £578.1$$

Belt Reinforcement Design Inspection Cost

$$C_{D,BeltReinf\mathrm{(INSP)}} = 26.04 \times 10 \times 1$$
$$= £260.4$$

Total Belt Reinforcement Design Cost

$$C_{BeltReinf} = £578.1 + £260.4$$
$$=£838.5$$

Window Channel

Inputs

Design hours – 60

Designer Rate – £19.27/hr

Inspection hours of – 5

Inspector Rate - £26.04/hr

Number of Designers – 1

Window Channel Design Cost

$$C_{D,WindChan} = 19.27 \times 60 \times 1$$
$$= £1156.2$$

Window Channel Design Cost

$$C_{D,WindChan(\mathrm{INSP})} = 26.04 \times 5 \times 1$$
$$= £130.2$$

Subtotal for Window Channel Design Cost was calculated using equation (3.7) is therefore

$$C_{WindChan} = £1156.2 + £130$$
$$= £1286.4$$

Using equation (3.1) the total RLW door design cost is

$$C_{T,pdt} = \sum(£4114.4 + £7853.8 + £10{,}312 + £901 + £1609.3 + £838.5 + £838.5 + £1286.4)$$
$$= £27753.9$$

4.4.2 RLW Process Model

The RLW process model consists of inputs required to build the models, modules of the RLW process and the expected outputs of the process model.

4.4.2.1 RLW Process Model Inputs

The inputs for the process model are;

1. RLW product sequence diagram
2. Workstation and operations cycle times
3. Workstations design cost estimation formulae

4.4.2.2 RLW Process Modules

The RLW Process Model follows the diagram shown in Figure 3.7 which shows the inputs, modules and output for RLW. Some of the inputs are generated from the product model such as the Product Sequence Diagram whereas others are introduced to generate the modules.

(1) RLW Process Illustration

RLW Workstation Sequence – the Remote Laser Welding process has 8 workstations named ST100 through to ST170 as shown in Figure 4.5.

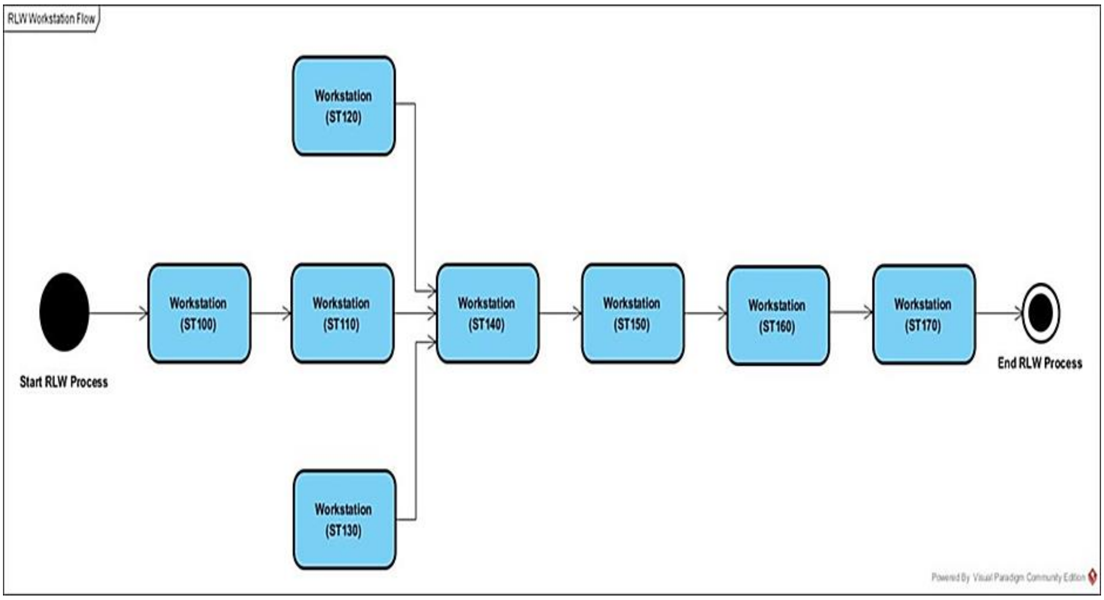


Figure 4. 6 RLW Process Illustration

The diagram shows the relationship between the workstations and how the welding process is finally completed in a static representation.

(2) ***RLW Workstations Sequence Representation***

Figure 4.6 is an XML format generated from the workstation sequence diagram, showing how a computer system read the workstation flow.

```
ConnectionPointStyle="0" AutoFitShapesSize="false" AlignToGrid="false">
 - <Shapes>
   + <InitialNode Name="Start RLW Process" ModelElementNameAlignment="9" Id="FBn8mVqESzVmxwam" Y="346" X="217" Width="113" ShowAllocatedTo="0" ShowAllocatedFrom="0" Height="71" ZOrder="19"
     ShowStereotypeIconName="0" Selectable="true" RequestSetSizeOption="0" RequestResetCaptionSize="false" RequestResetCaptionFitWidth="false" RequestResetCaption="false" RequestFitSizeFromCenter="false"
     RequestFitSize="false" RequestDefaultSize="false" PrimitiveShapeType="0" PresentationOption="4" ParentConnectorLineLength="10" ParentConnectorHeaderLength="40" ParentConnectorDTheta="0.0"
     OverrideAppearanceWithStereotypeIcon="true" Model="TBn8mVqESzVmxwbS" MetaModelElement="TBn8mVqESzVmxwbS" Foreground="rgb(0, 0, 0)" CreatorDiagramType="ActivityDiagram"
     ConnectionPointType="1" ConnectToPoint="true" Background="rgb(122, 207, 245)">
   + <ActivityAction Name="Workstation (ST100)" ModelElementNameAlignment="9" Id="xBMCmVqESzVmxwbc" Y="338" X="393" Width="145" ShowAllocatedTo="0" ShowAllocatedFrom="0" Height="88"
     ZOrder="17" ShowStereotypeIconName="0" Selectable="true" RequestSetSizeOption="0" RequestResetCaptionSize="false" RequestResetCaptionFitWidth="false" RequestResetCaption="false"
     RequestFitSizeFromCenter="false" RequestFitSize="false" RequestDefaultSize="false" PrimitiveShapeType="0" PresentationOption="4" ParentConnectorLineLength="10" ParentConnectorHeaderLength="40"
     ParentConnectorDTheta="0.0" OverrideAppearanceWithStereotypeIcon="true" Model="xBMCmVqESzVmxwbd" MetaModelElement="xBMCmVqESzVmxwbd" Foreground="rgb(0, 0, 0)" ConnectionPointType="2"
     ConnectToPoint="true" Background="rgb(122, 207, 245)" ShowCallBehaviorOption="0">
   + <ActivityAction Name="Workstation (ST120)" ModelElementNameAlignment="9" Id="itMCmVqESzVmxwbl" Y="151" X="599" Width="145" ShowAllocatedTo="0" ShowAllocatedFrom="0" Height="88" ZOrder="15"
     ShowStereotypeIconName="0" Selectable="true" RequestSetSizeOption="0" RequestResetCaptionSize="false" RequestResetCaptionFitWidth="false" RequestResetCaption="false" RequestFitSizeFromCenter="false"
     RequestFitSize="false" RequestDefaultSize="false" PrimitiveShapeType="0" PresentationOption="4" ParentConnectorLineLength="10" ParentConnectorHeaderLength="40" ParentConnectorDTheta="0.0"
     OverrideAppearanceWithStereotypeIcon="true" Model="StMCmVqESzVmxwbm" MetaModelElement="StMCmVqESzVmxwbm" Foreground="rgb(0, 0, 0)" ConnectionPointType="2" ConnectToPoint="true"
     Background="rgb(122, 207, 245)" ShowCallBehaviorOption="0">
   + <ActivityAction Name="Workstation (ST140)" ModelElementNameAlignment="9" Id="U8sCmVqESzVmxwbu" Y="338" X="804" Width="145" ShowAllocatedTo="0" ShowAllocatedFrom="0" Height="88"
     ZOrder="13" ShowStereotypeIconName="0" Selectable="true" RequestSetSizeOption="0" RequestResetCaptionSize="false" RequestResetCaptionFitWidth="false" RequestResetCaption="false"
     RequestFitSizeFromCenter="false" RequestFitSize="false" RequestDefaultSize="false" PrimitiveShapeType="0" PresentationOption="4" ParentConnectorLineLength="10" ParentConnectorHeaderLength="40"
     ParentConnectorDTheta="0.0" OverrideAppearanceWithStereotypeIcon="true" Model="U8sCmVqESzVmxwbv" MetaModelElement="U8sCmVqESzVmxwbv" Foreground="rgb(0, 0, 0)" ConnectionPointType="2"
     ConnectToPoint="true" Background="rgb(122, 207, 245)" ShowCallBehaviorOption="0">
   + <ActivityAction Name="Workstation (ST130)" ModelElementNameAlignment="9" Id="wGsCmVqESzVmxwb3" Y="524" X="600" Width="145" ShowAllocatedTo="0" ShowAllocatedFrom="0" Height="88"
     ZOrder="11" ShowStereotypeIconName="0" Selectable="true" RequestSetSizeOption="0" RequestResetCaptionSize="false" RequestResetCaptionFitWidth="false" RequestResetCaption="false"
     RequestFitSizeFromCenter="false" RequestFitSize="false" RequestDefaultSize="false" PrimitiveShapeType="0" PresentationOption="4" ParentConnectorLineLength="10" ParentConnectorHeaderLength="40"
     ParentConnectorDTheta="0.0" OverrideAppearanceWithStereotypeIcon="true" Model="IGsCmVqESzVmxwb4" MetaModelElement="IGsCmVqESzVmxwb4" Foreground="rgb(0, 0, 0)" ConnectionPointType="2"
     ConnectToPoint="tr 7, 245)" ShowCallBehaviorOption="0">
   + <ActivityAction Nam ="Workstation (ST110)" M delElementNameAlignment="9" Id="eBsCmVqESzVmxwcA" Y="338" X="597" Width="145" ShowAllocatedTo="0" ShowAllocatedFrom="0" Height="88" ZOrder="8"
     ShowStereotypeIcon RequestSetSizeOption="0" RequestResetCaptionSize="false" RequestResetCaptionFitWidth="false" RequestResetCaption="false" RequestFitSizeFromCenter="false"
     RequestFitSize="false" RequestDefaultSize="false" PrimitiveShapeType="0" PresentationOption="4" ParentConnectorLineLength="10" ParentConnectorHeaderLength="40" ParentConnectorDTheta="0.0"
     OverrideAppearanceWithStereotypeIcon="true" Model=".BsCmVqESzVmxwcB" MetaModelElement=".BsCmVqESzVmxwcB" Foreground="rgb(0, 0, 0)" ConnectionPointType="2" ConnectToPoint="true"
     Background="rgb(122, 207, 245)" ShowCallBehaviorOption="0">
   + <ActivityAction Name="Workstation (ST150)" ModelElementNameAlignment="9" Id="QFsCmVqESzVmxwcK" Y="338" X="1011" Width="145" ShowAllocatedTo="0" ShowAllocatedFrom="0" Height="88"
```


Figure 4. 7: RLW Workstations Sequence Representation

This is required when executing the process sequence in the cost modeller later in the methodology. The complete script is shown in Appendix D.1. Changes made on the static diagram is automatically updated in the computer representation.

ST110 Operation Sequence – each workstation's task is broken down into operations to plan activities the must be done within the workstation.

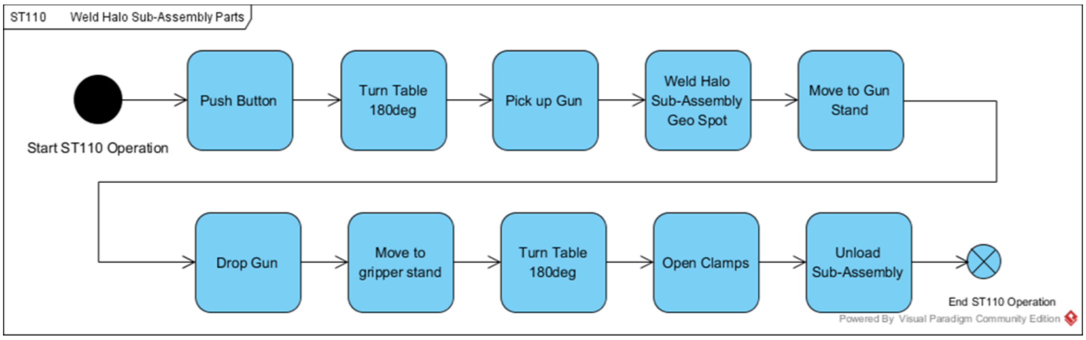


Figure 4. 8: ST110 Operation Sequence

Operational activities for workstation ST110 is shown in Figure4.7 and the rest of the workstations' operations are displayed in Appendix E.1 to E.8.

RLW ST110 Sequence Representation – the operation sequence representation is generated from the operation sequence in XML format to show how computer systems understand and interpret the flow of sequence as shown in Figure 4.8.

```
+ <MasterView>
- <ModelChildren>
  + <InitialNode Name="Start ST110 Operation" Id="sxUHyWqESzVmxz7O" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-09-09T13:31:13.057" PmCreateDateTime="2016-08-16T16:03:09.261" PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" MustIsolate="false" Leaf="false">
  + <ActivityAction Name="Push Button" Id="pLInyWqESzVmxz7h" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-09-09T13:31:13.057" PmCreateDateTime="2016-08-16T16:07:07.941" PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" MustIsolate="false" Leaf="false">
  + <ActivityAction Name="Turn Table 180deg" Id="6_InyWqESzVmxz7q" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-09-09T13:31:13.057" PmCreateDateTime="2016-08-16T16:07:08.631" PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" MustIsolate="false" Leaf="false">
  + <ActivityAction Name="Weld Halo Sub-Assembly Geo Spot" Id="6konyWqESzVmxz7z" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-09-09T13:31:13.057" PmCreateDateTime="2016-08-16T16:07:09.271" PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" MustIsolate="false" Leaf="false">
  + <ActivityAction Name="Pick up Gun" Id="IwvXyWqESzVmxz8j" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-09-09T13:31:13.057" PmCreateDateTime="2016-08-16T16:15:20.388" PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" MustIsolate="false" Leaf="false">
  + <ActivityAction Name="Move to Gun Stand" Id="J8vXyWqESzVmxz8s" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-09-09T13:31:13.057" PmCreateDateTime="2016-08-16T16:15:21.188" PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" MustIsolate="false" Leaf="false">
  + <ActivityAction Name="Drop Gun" Id=".2QPyWqESzVmxz9w" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-09-09T13:31:13.057" PmCreateDateTime="2016-08-16T16:20:04.575" PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" MustIsolate="false" Leaf="false">
  + <ActivityAction Name="Move to gripper stand" Id="UeQPyWqESzVmxz95" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-09-09T13:31:13.057" PmCreateDateTime="2016-08-16T16:20:04.746" PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" MustIsolate="false" Leaf="false">
  + <ActivityAction Name="Turn Table 180deg" Id="bUwPyWqESzVmxz.C" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-09-09T13:31:13.057" PmCreateDateTime="2016-08-16T16:20:07.606" PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" MustIsolate="false" Leaf="false">
  + <ActivityAction Name="Unload Sub-Assembly" Id="QSwPyWqESzVmxz.L" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-09-09T13:31:13.057" PmCreateDateTime="2016-08-16T16:20:08.066" PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" MustIsolate="false" Leaf="false">
  + <ActivityAction Name="Open Clamps" Id="T.wPyWqESzVmxz.d" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-09-09T13:31:13.057" PmCreateDateTime="2016-08-16T16:20:08.946" PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" MustIsolate="false" Leaf="false">
  + <FlowFinalNode Name="End ST110 Operation" Id="wGSfyWqESzVmx0D0" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-09-09T13:31:13.057" PmCreateDateTime="2016-08-16T16:29:54.179" PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" MustIsolate="false" Leaf="false">
  </ModelChildren>
</Activity>
```


Figure 4. 9: RLW ST110 Sequence Representation

For full computer representation for all Operations Sequence see Appendix D.1

(3) RLW Process Design Cost Calculation

The process design cost for RLW only considers the workstation design cost as it assumes that the workstation design also includes the operation of the workstations. However, for a more complicated process, the design of the process operations may have different cost values depending on how long it takes the process designer to design all the workstations.

The inputs required to estimate the design cost of the RLW production process are given as:

Inputs

Workstation design time = T_{ST}

$Workstation\ Designer's\ Rate = R_{WD}$

Workstation Inspection time = $T_{W(INSP)}$

$Workstation\ Inspection\ Rate = R_{W(INSP)}$

Process designer's annual salary = £42000

Process Inspector's annual salary = £52000

Using equation (3.10) the workstation designer's rate is calculated as:

$$R_{WD} = \frac{£42000}{40hrsX48wks} = 21.87$$

$$\cong £22/hr$$

The inspector's rate is also calculated using equation (3.12) as

$$C_{W(INSP)} = \frac{£52000}{40hrsX48wks}$$

$$\cong £27$$

Therefore the cost of designing the workstation is calculated using equation (3.9) as

$$C_{ST} = T_{ST} * R_{WD}$$

$$C_{ST100} = 24 * 22 = 528$$

$$C_{ST110} = 40 * 22 = 880$$

$$C_{ST120} = 21 * 22 = 462$$

$$C_{ST130} = 40 * 22 = 880$$

$$C_{ST140} = 48 * 22 = 1056$$

$$C_{ST150} = 8 * 22 = 176$$

$$C_{ST160} = 16 * 22 = 352$$

$$C_{ST170} = 16 * 22 = 352$$

$$C_{ST} = \sum(528 + 880 + 462 + 880 + 1056 + 176 + 352 + 352) = £4,686$$

$$C_{ProcInsp} = Inspector\ Rate * Inspection\ Time = 27 * 40 = £1,080$$

Using equation (3.13) the total cost of all the workstations design is calculated as:

$$C_{ProcD} = C_{W(INSP)} + Total\ C_{ST}$$

$$= 4686 + 1080$$

$$= £5,766$$

Workstation design tool cost is not considered in this cost equation as existing tools were used.

4.4.2.3 RLW Process Output

The outputs of the RLW process model are:

- Graphical illustration of the RLW process sequence that shows the relationship between all the workstations with detailed operations for each workstation
- A computer representation of both process and workstations' sequence for RLW
- RLW process design cost calculations

4.4.3 RLW Resource Model

4.4.3.1 RLW Resource Model Input

Inputs for the resource model are Resource Design Cost Estimation Formulae and the Workstations Sequence Diagram.

4.4.3.2 RLW Resource Module

Figure 4.9 Shows the modules developed for the RLW resources with the inputs and outputs of the model.

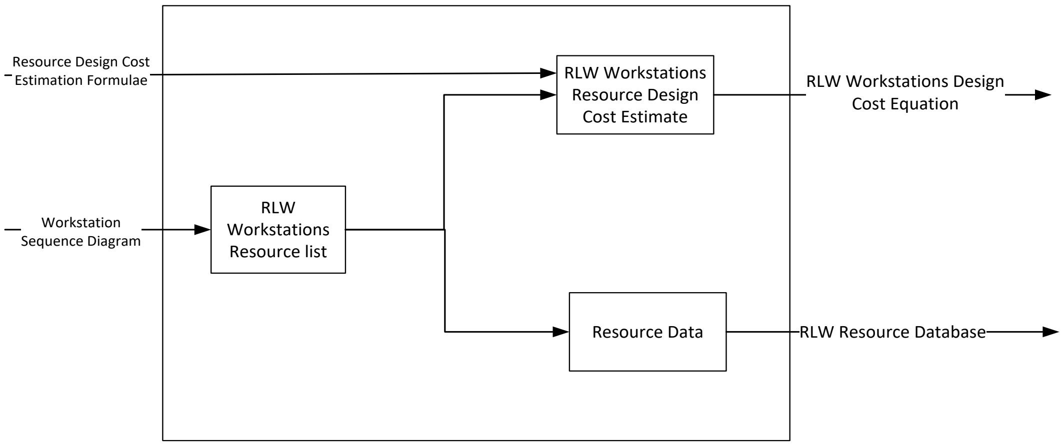


Figure 4. 10: RLW Resource Model

(1) RLW Workstations Resource List

Resources required for the remote laser welding of the car door are shown in Figure 4.10. The resource list for this case application can further be expanded to shows the roles and functions performed by each resource.

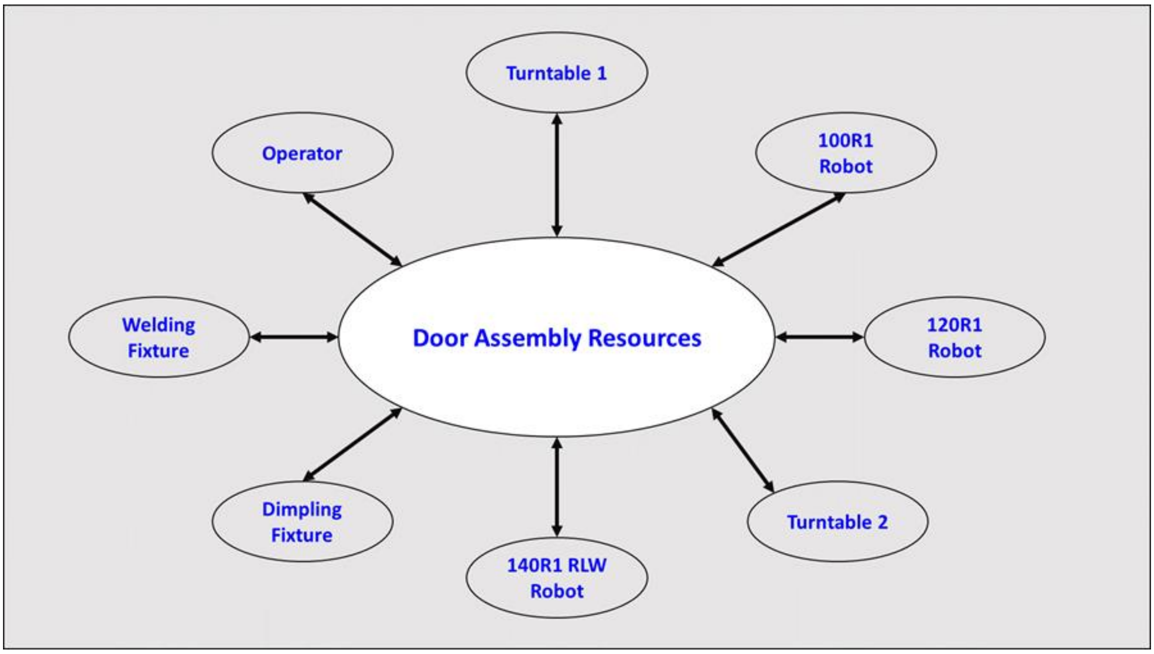


Figure 4. 11: RLW Resource Model

Resources for the RLW process accomplishes tasks such as pick and place by robots; turn parts by fixtures; laser welding by robots and loading and pushing buttons by the operator. For example, the RLW operator's function is shown in Figure 4.11, illustrating major activities performed by the operator. For all other resource activities, please see Appendix F.1 to F.11.

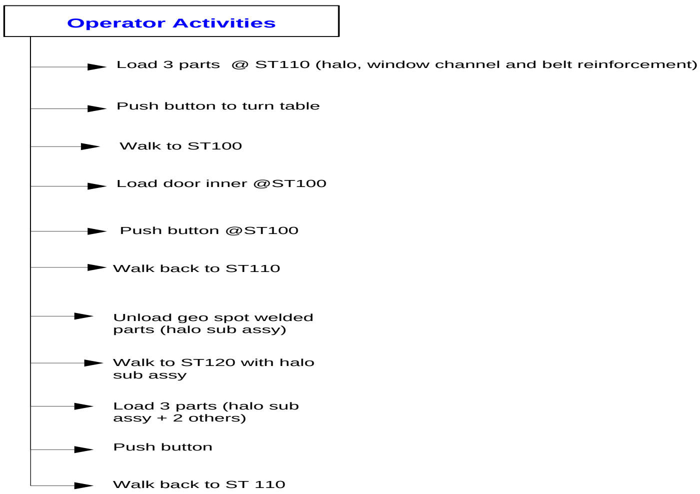


Figure 4. 12: Operator Activity

(2) RLW Resource Data

Table 4.1 is a screenshot of the resource database created for the RLW process detailing the various resource equipment within their various workstations.

Table 4. 1: RLW Resource Database

Work Station	Resources	Resource Detail	Cost of Resource	Resource Quantity	Cost per Operation
OP 110	Rise and Fall Guard	Shutter Guard for safety, start and stop of operat	£5,000	1	£5,000
	RSW Gun Tool Stand		£5,000	1	£5,000
	Change plate		£5,000	1	£5,000
	Gun + weld timer		£20,000	1	£20,000
	Turn Table for OP110		£12,000	1	£12,000
	Turn table Fixture	Automated- industrial	£15,000	2	£30,000
					£77,000
OP 120	120R1 Grippper Robot		£45,000	1	£45,000
	Gripper-double		£20,000	1	£20,000
	Robot slidding Rail		£40,000	1	£40,000
	Robot Base	Mounting point and screws	£2,000	1	£2,000
	Double Putdown Fixture Table		£10,000	2	£20,000
	Rise and Fall Guard	Shutter Guard for safety, start and stop of operat	£5,000	1	£5,000
					£132,000
OP 140 RLW	Laser	IPG 4kW fibre laser incl. smart laser and water chi	£200,000	0.5	£100,000
	Chiller		£30,000	0.5	£15,000
	Dimpling Fixture	Automated- industrial	£30,000	2	£60,000
	Welding Fixture	Semi automatic welding fixture, supplied by COM	£100,000	2	£200,000
	Robot Base	Mounting point and screws	£2,000	0.5	£1,000
	Robot Arm	6 axis COMAU robot arm and controls	£125,000	0.5	£62,500
	Delivery Fibre	20m length, 200µm diameter, IPG connector	£3,000	0.5	£1,500
	Cell enclosure	Laser safe cell and beam monitoring, access door, CCTV,	£60,000	0.5	£30,000
	Turn table for RLW (2 stations)		£20,000.00	1	£20,000
					£490,000

(3) RLW Resource Design Cost calculation

Using equation:

$$C_{RD,m} = \sum_{m \geq 1}^{n} (R_{RD} \times T_{RD} \times N_{RD})$$

Where

R_{RD} = Resource Designer's Rate

T_{RD} = Resource Designers' Time = 120hr

N_{RD} = Number of Resource Designers = 1

C_{RD} = Resource Design Cost

$C_{ResInsp}$ = Cost of Resource Design Inspection

Also,

$$R_{RD} = \frac{Annual\ Salary\ of\ Resource\ Designer}{Expected\ Annual\ Working\ Hours}$$

$$R_{RD} = \frac{35000}{40hrsX48wks}$$

$$= £18.23/Hr$$

Therefore,

$$C_{RD,m} = (18.23 \times 120 \times 1)$$
$$= £2187.6$$
$$C_{ResInsp} = Inspector\ Rate * Inspection\ Time = 27 * 24 = £648$$
$$Total\ Resource\ Design\ Cost = R_{RD} + C_{ResInsp}$$
$$= £2187.6 + £648$$
$$\boldsymbol{Total\ Resource\ Design\ Cost} = £\mathbf{2835.6}$$

4.4.3.3 RLW Resource Model Output

The outputs are structured resource database for RLW resources with cost information and the RLW Resources design cost estimate with cost equations.

4.4.4 RLW Process and Resource Integration Model

4.4.4.1 RLW Process and Resource Integration Inputs

The inputs for process and Resources integration are Workstations Sequence Diagram, Operation Sequence Diagram, Resource Database and Cost Estimation Formulae.

4.4.4.2 RLW Process and Resource Integration Modules

(1) RLW Process and Resource Integration Illustration

The integration of RLW process and resources is illustrated in Figure…. graphically where RLW Resources are Assigned to RLW workstations.

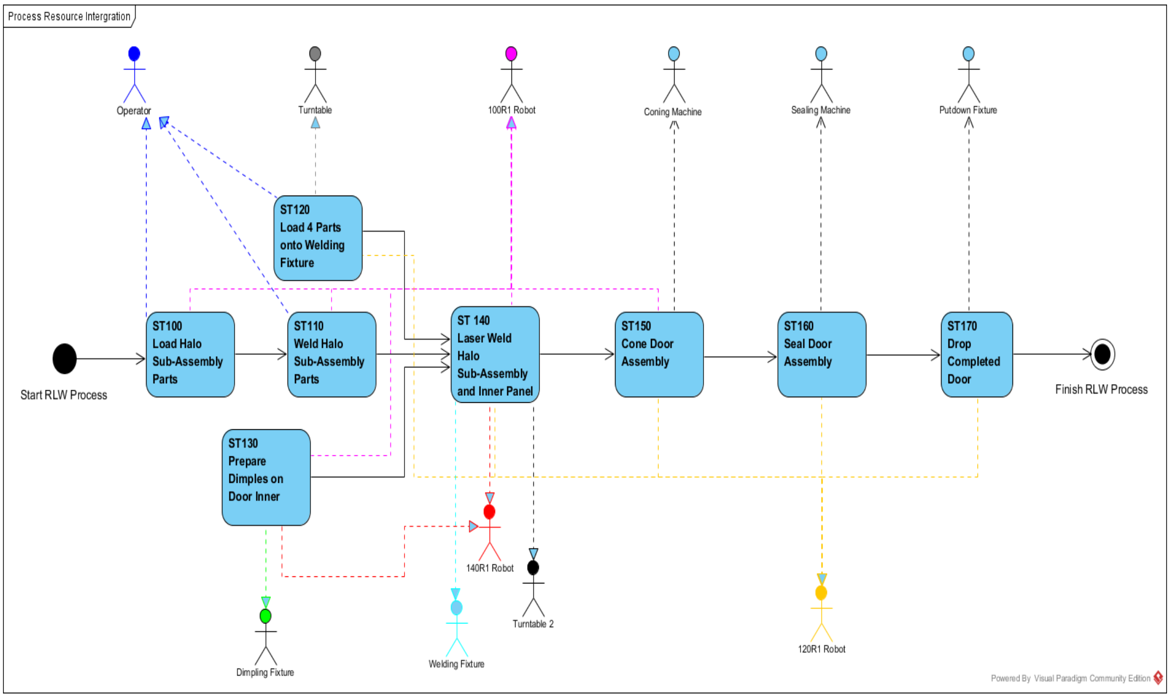


Figure 4. 13: RLW Process and Resource Integration Illustration

Table 4.2 further illustrated a matrix of the relationship between workstations and resources assigned to them.

Table 4. 2: Process and Resource Matrix

	Process and Resource Matrix								
		Operator	Turntable 1	100R1 Robot	120R1 Robot	Turntable 2	140R1 Robot	Dimpling Fixture	Welding Fixture
Workstations	ST100	√		√					
	ST110	√		√					
	ST120	√	√		√				
	ST130						√	√	
	ST140			√	√		√		√
	ST150			√	√	√			
	ST160				√				
	ST170				√				
	Resources								

However, the integration is also done at the workstation level of the RLW process, where each workstation's activity shows the resources acting on them. Figure 4.13 shows workstation ST110's integrational activities. The rest of the workstations' integration are displayed in Appendix D.2 to D.8.

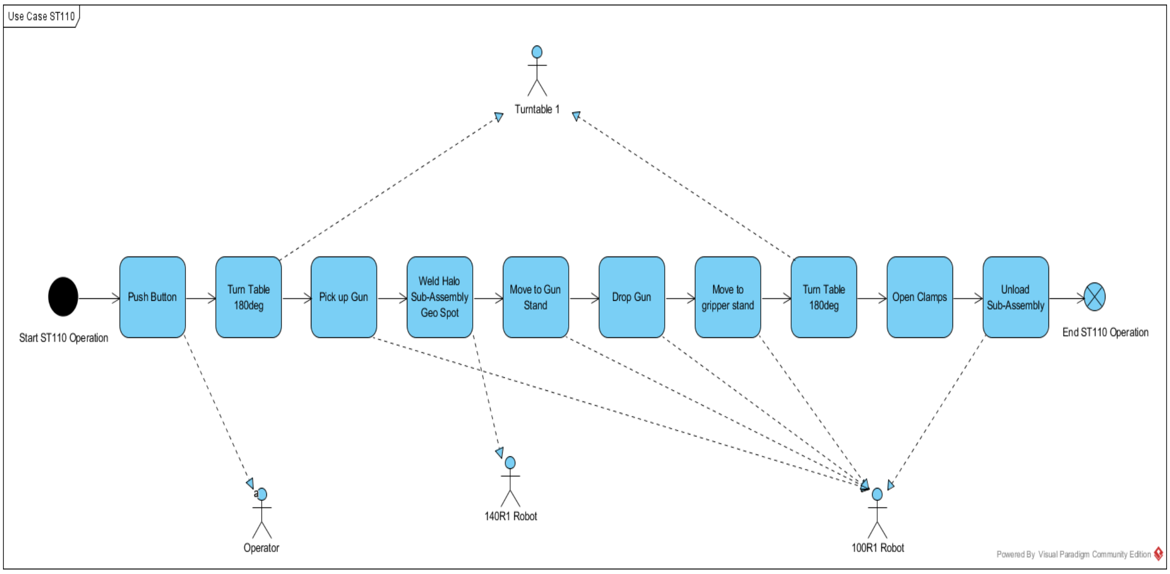
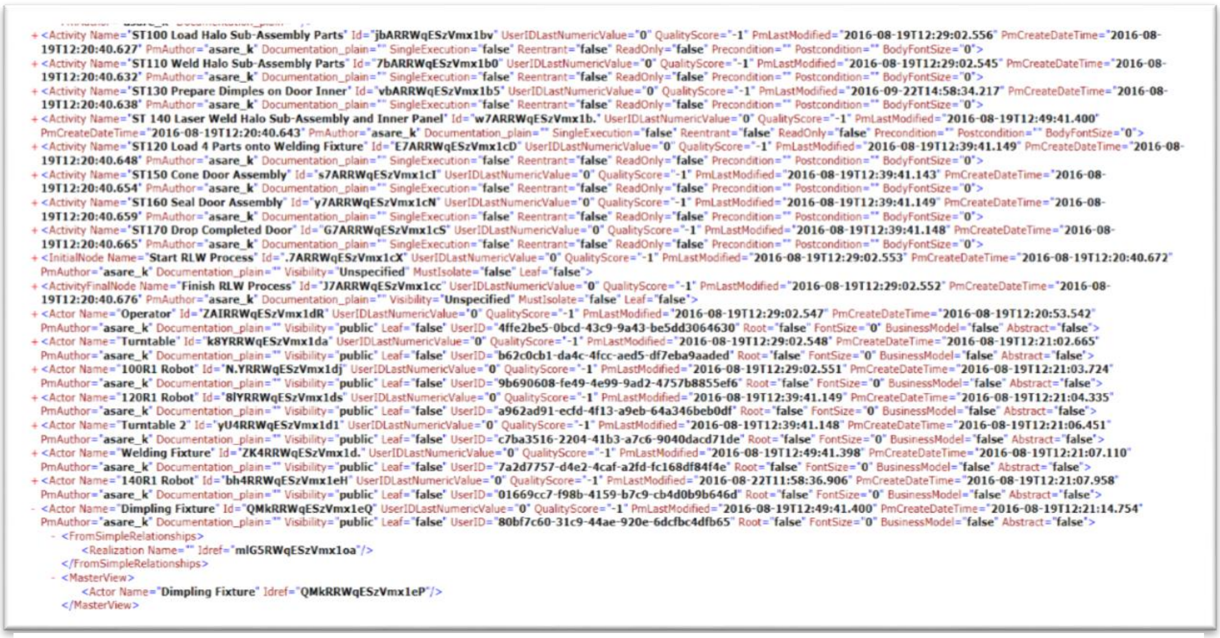


Figure 4. 14: ST110 Operation and Resource Activities

(2) RLW Process and Resource Integration Representation

The RLW process and resource integration is represented in Figure 4.14. This representation is generated to show how a computer system can understand the RLW process and resource integration.

Workstation and resource integration for ST110 is also represented in Figure 4.15.



```
+ <Activity Name="ST100 Load Halo Sub-Assembly Parts" Id="jbARRWqESzVmx1bv" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-08-19T12:29:02.556" PmCreateDateTime="2016-08-19T12:20:40.627" PmAuthor="asare_k" Documentation_plain="" SingleExecution="false" Reentrant="false" ReadOnly="false" Precondition="" Postcondition="" BodyFontSize="0">
+ <Activity Name="ST110 Weld Halo Sub-Assembly Parts" Id="7bARRWqESzVmx1b0" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-08-19T12:29:02.545" PmCreateDateTime="2016-08-19T12:20:40.632" PmAuthor="asare_k" Documentation_plain="" SingleExecution="false" Reentrant="false" ReadOnly="false" Precondition="" Postcondition="" BodyFontSize="0">
+ <Activity Name="ST130 Prepare Dimples on Door Inner" Id="vbARRWqESzVmx1b5" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-09-22T14:58:34.217" PmCreateDateTime="2016-08-19T12:20:40.638" PmAuthor="asare_k" Documentation_plain="" SingleExecution="false" Reentrant="false" ReadOnly="false" Precondition="" Postcondition="" BodyFontSize="0">
+ <Activity Name="ST 140 Laser Weld Halo Sub-Assembly and Inner Panel" Id="w7ARRWqESzVmx1b." UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-08-19T12:49:41.400" PmCreateDateTime="2016-08-19T12:20:40.643" PmAuthor="asare_k" Documentation_plain="" SingleExecution="false" Reentrant="false" ReadOnly="false" Precondition="" Postcondition="" BodyFontSize="0">
+ <Activity Name="ST120 Load 4 Parts onto Welding Fixture" Id="E7ARRWqESzVmx1cD" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-08-19T12:39:41.149" PmCreateDateTime="2016-08-19T12:20:40.648" PmAuthor="asare_k" Documentation_plain="" SingleExecution="false" Reentrant="false" ReadOnly="false" Precondition="" Postcondition="" BodyFontSize="0">
+ <Activity Name="ST150 Cone Door Assembly" Id="s7ARRWqESzVmx1cI" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-08-19T12:39:41.143" PmCreateDateTime="2016-08-19T12:20:40.654" PmAuthor="asare_k" Documentation_plain="" SingleExecution="false" Reentrant="false" ReadOnly="false" Precondition="" Postcondition="" BodyFontSize="0">
+ <Activity Name="ST160 Seal Door Assembly" Id="y7ARRWqESzVmx1cN" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-08-19T12:39:41.149" PmCreateDateTime="2016-08-19T12:20:40.659" PmAuthor="asare_k" Documentation_plain="" SingleExecution="false" Reentrant="false" ReadOnly="false" Precondition="" Postcondition="" BodyFontSize="0">
+ <Activity Name="ST170 Drop Completed Door" Id="G7ARRWqESzVmx1cS" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-08-19T12:39:41.148" PmCreateDateTime="2016-08-19T12:20:40.665" PmAuthor="asare_k" Documentation_plain="" SingleExecution="false" Reentrant="false" ReadOnly="false" Precondition="" Postcondition="" BodyFontSize="0">
+ <InitialNode Name="Start RLW Process" Id=".7ARRWqESzVmx1cX" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-08-19T12:29:02.553" PmCreateDateTime="2016-08-19T12:20:40.672" PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" MustIsolate="false" Leaf="false">
+ <ActivityFinalNode Name="Finish RLW Process" Id="J7ARRWqESzVmx1cc" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-08-19T12:29:02.552" PmCreateDateTime="2016-08-19T12:20:40.676" PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" MustIsolate="false" Leaf="false">
+ <Actor Name="Operator" Id="ZAIRRWqESzVmx1dR" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-08-19T12:29:02.547" PmCreateDateTime="2016-08-19T12:20:53.542" PmAuthor="asare_k" Documentation_plain="" Visibility="public" Leaf="false" UserID="4ffe2be5-0bcd-43c9-9a43-be5dd3064630" Root="false" FontSize="0" BusinessModel="false" Abstract="false">
+ <Actor Name="Turntable" Id="k8YRRWqESzVmx1da" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-08-19T12:29:02.548" PmCreateDateTime="2016-08-19T12:21:02.665" PmAuthor="asare_k" Documentation_plain="" Visibility="public" Leaf="false" UserID="b62c0cb1-da4c-4fcc-aed5-df7eba9aaded" Root="false" FontSize="0" BusinessModel="false" Abstract="false">
+ <Actor Name="100R1 Robot" Id="N.YRRWqESzVmx1dj" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-08-19T12:29:02.551" PmCreateDateTime="2016-08-19T12:21:03.724" PmAuthor="asare_k" Documentation_plain="" Visibility="public" Leaf="false" UserID="9b690608-fe49-4e99-9ad2-4757b8855ef6" Root="false" FontSize="0" BusinessModel="false" Abstract="false">
+ <Actor Name="120R1 Robot" Id="8lYRRWqESzVmx1ds" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-08-19T12:39:41.149" PmCreateDateTime="2016-08-19T12:21:04.335" PmAuthor="asare_k" Documentation_plain="" Visibility="public" Leaf="false" UserID="a962ad91-ecfd-4f13-a9eb-64a346beb0df" Root="false" FontSize="0" BusinessModel="false" Abstract="false">
+ <Actor Name="Turntable 2" Id="yU4RRWqESzVmx1d1" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-08-19T12:39:41.148" PmCreateDateTime="2016-08-19T12:21:06.451" PmAuthor="asare_k" Documentation_plain="" Visibility="public" Leaf="false" UserID="c7ba3516-2204-41b3-a7c6-9040dacd71de" Root="false" FontSize="0" BusinessModel="false" Abstract="false">
+ <Actor Name="Welding Fixture" Id="ZK4RRWqESzVmx1d." UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-08-19T12:49:41.398" PmCreateDateTime="2016-08-19T12:21:07.110" PmAuthor="asare_k" Documentation_plain="" Visibility="public" Leaf="false" UserID="7a2d7757-d4e2-4caf-a2fd-fc168df84f4e" Root="false" FontSize="0" BusinessModel="false" Abstract="false">
+ <Actor Name="140R1 Robot" Id="bh4RRWqESzVmx1eH" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-08-22T11:58:36.906" PmCreateDateTime="2016-08-19T12:21:07.958" PmAuthor="asare_k" Documentation_plain="" Visibility="public" Leaf="false" UserID="01669cc7-f98b-4159-b7c9-cb4d0b9b646d" Root="false" FontSize="0" BusinessModel="false" Abstract="false">
- <Actor Name="Dimpling Fixture" Id="QMkRRWqESzVmx1eQ" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-08-19T12:49:41.400" PmCreateDateTime="2016-08-19T12:21:14.754" PmAuthor="asare_k" Documentation_plain="" Visibility="public" Leaf="false" UserID="80bf7c60-31c9-44ae-920e-6dcfbc4dfb65" Root="false" FontSize="0" BusinessModel="false" Abstract="false">
  - <FromSimpleRelationships>
      <Realization Name="" Idref="mlG5RWqESzVmx1oa"/>
    </FromSimpleRelationships>
  - <MasterView>
      <Actor Name="Dimpling Fixture" Idref="QMkRRWqESzVmx1eP"/>
    </MasterView>
```


Figure 4. 15: RLW Process and Resource Integration Representation (Workstation)

```
- <ModelChildren>
  + <InitialNode Name="Start ST110 Operation" Id="sxUHyWqESzVmxz70" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-09-09T13:31:13.057" PmCreateDateTime="2016-08-
    16T16:03:09.261" PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" MustIsolate="false" Leaf="false">
  + <ActivityAction Name="Push Button" Id="pLInyWqESzVmxz7h" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-09-09T13:31:13.057" PmCreateDateTime="2016-08-16T16:07:07.94
    PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" MustIsolate="false" Leaf="false">
  + <ActivityAction Name="Turn Table 180deg" Id="6_InyWqESzVmxz7q" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-09-09T13:31:13.057" PmCreateDateTime="2016-08-
    16T16:07:08.631" PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" MustIsolate="false" Leaf="false">
  + <ActivityAction Name="Weld Halo Sub-Assembly Geo Spot" Id="6konyWqESzVmxz7z" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-09-09T13:31:13.057" PmCreateDateTime="201
    08-16T16:07:09.271" PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" MustIsolate="false" Leaf="false">
  + <ActivityAction Name="Pick up Gun" Id="IwvXyWqESzVmxz8j" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-09-09T13:31:13.057" PmCreateDateTime="2016-08-16T16:15:20.38
    PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" MustIsolate="false" Leaf="false">
  + <ActivityAction Name="Move to Gun Stand" Id="J8vXyWqESzVmxz8s" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-09-09T13:31:13.057" PmCreateDateTime="2016-08-
    16T16:15:21.188" PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" MustIsolate="false" Leaf="false">
  + <ActivityAction Name="Drop Gun" Id=".2QPyWqESzVmxz9w" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-09-09T13:31:13.057" PmCreateDateTime="2016-08-16T16:20:04.575"
    PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" MustIsolate="false" Leaf="false">
  + <ActivityAction Name="Move to gripper stand" Id="UeQPyWqESzVmxz95" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-09-09T13:31:13.057" PmCreateDateTime="2016-08-
    16T16:20:04.746" PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" MustIsolate="false" Leaf="false">
  + <ActivityAction Name="Turn Table 180deg" Id="bUwPyWqESzVmxz.C" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-09-09T13:31:13.057" PmCreateDateTime="2016-08-
    16T16:20:07.606" PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" MustIsolate="false" Leaf="false">
  + <ActivityAction Name="Unload Sub-Assembly" Id="QSwPyWqESzVmxz.L" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-09-09T13:31:13.057" PmCreateDateTime="2016-08-
    16T16:20:08.066" PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" MustIsolate="false" Leaf="false">
  + <ActivityAction Name="Open Clamps" Id="T.wPyWqESzVmxz.d" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-09-09T13:31:13.057" PmCreateDateTime="2016-08-16T16:20:08.94
    PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" MustIsolate="false" Leaf="false">
  + <FlowFinalNode Name="End ST110 Operation" Id="wGSfyWqESzVmx0D0" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-09-09T13:31:13.057" PmCreateDateTime="2016-08-
    16T16:29:54.179" PmAuthor="asare_k" Documentation_plain="" Visibility="Unspecified" MustIsolate="false" Leaf="false">
  </ModelChildren>
</Activity>
<ModelRelationshipContainer Name="relationships" Id="WJ5VyWqESzVmxzvI" UserIDLastNumericValue="0" QualityScore="-1" PmLastModified="2016-10-03T14:34:02.386" PmCreateDateTime="2016-08-
16T15:03:58.106" PmAuthor="asare_k" Documentation_plain="">
lodels>
agrams>
```


Figure 4. 16: Workstation ST110 and Resource Integration

(3) ***RLW Process and Resource Integration Cost Calculation***

Integrating the RLW Process and its Resources generates the cost of the system. This cost calculation considers the cost of human operator resources rates and machine resource rates. RLW machine resources depreciation and rate are shown in Table 4.3 using:

$$Machine\ Resources\ Rate\ (R_{Rm}) = \frac{Purchase\ Price - Depreciation}{Predicted\ Annual\ Hours\ of\ Operation} \tag{4.1}$$

Depreciation is calculated using the straight-line method, assuming that the useful life of machine resources is 10 years at which its salvage value will be zero.

Therefore

$$Depreciation = \frac{(cost\ of\ Asset - Salvage\ Value)}{Estimated\ Useful\ Life} \tag{4.2}$$

Table 4.3 shows the workstations, their rates, cycle time and the cost incurred through the consumption of resources within the workstations. Totaling the workstation’s cost gives the cost estimate for making 72 welds using the Remote Laser Welding technology.

Table 4. 3: RLW Resources Rates

RLW Resources Rates					
Resource Type	**Purchase Price (£)**	**Useful Life (Yrs)**	**Salvage Value**	**Depreciation**	**Resource Rate (£/hr)**
Turntable 1	42000	10	0	4200	19.6875
Turntable 2	20000	10	0	2000	9.375
100R1 Robot	57500	10	0	5750	26.953125
120R1 Robot	10700	10	0	1070	5.015625
140R1 Robot	420000	10	0	42000	196.875
Dimpling Fixture	60000	10	0	6000	28.125
Welding Fixture	200000	10	0	20000	93.75
coning Machine	20000	10	0	2000	9.375
Sealing Machine	50000	10	0	5000	23.4375
Putdown Fixture	15000	10	0	1500	7.03125

The human operator hourly rate cost calculation considers the annual salary of the operator and the expected annual working hours.

Therefore,

$$Operator\ Rate = \frac{Operator's\ annual\ salary}{Operator's\ expected\ annual\ hours}$$

$$= \frac{£25{,}000}{40hr \times 48wks}$$

$$= £13/hr$$

Table 4. 4: Workstations Cost Calculations

Work Stations	Resources	Resource Rates (£/Hr)	Cycle Times (Sec)	Cycle Times (hr)	Work Stations Cost (Cycle Time*Resource Rate)	Total Work Station Cost
ST100	Operator	13	35	0.009722222	0.126388889	0.388402778
	100R1 Robot	26.95			0.262013889	
ST110	Operator	13	40	0.011111111	0.144444444	0.443888889
	100R1 Robot	26.95			0.299444444	
ST120	Operator	13	28	0.007777778	0.101111111	0.254255556
	Turntable	19.69			0.153144444	
ST130	Dimpling Fixture	28.13	115	0.031944444	0.898597222	8.048722222
	100R1 Robot	26.95			0.860902778	
	140R1 Robot	196.88			6.289222222	
ST140	100R1 Robot	26.95	112	0.031111111	0.838444444	10.17208889
	140R1 Robot	196.88			6.125155556	
	Welding Fixture	93.75			2.916666667	
	Turntable 2	9.38			0.291822222	
ST150	100R1 Robot	26.95	20	0.005555556	0.149722222	0.229944444
	120R1 Robot	5.06			0.028111111	
	Coning Machine	9.38			0.052111111	
ST160	Sealing Machine	23.44	30	0.008333333	0.195333333	0.237166667
	120R1 Robot	5.02			0.041833333	
ST170	Putdown Fixture	7.03	40	0.011111111	0.078111111	0.133888889
	120R1 Robot	5.02			0.055777778	
Total Process Cost						**19.90835833**

4.4.4.3 RLW Process and Resource Integration Output

The output of the RLW process and resource integration are:

- Graphical illustrations of Process-Resource integration and detailed workstations-Resource integration
- A computer representation of both Process-Resource integration and detailed workstations-Resource integration
- Process and Resource Integration Design Cost Calculations

4.4.5 RLW's PPR Design Cost Calculator

The RLW PPR design cost calculator integrates the Product-Process-Resource design costs in a common database accessible to both engineers and manufacturing process designers as shown graphically in Figure 4.16. Due to the interactive nature of the design cost calculator, changes made to design cost parameters at any design stage automatically shows cost causalities in the total design cost values.

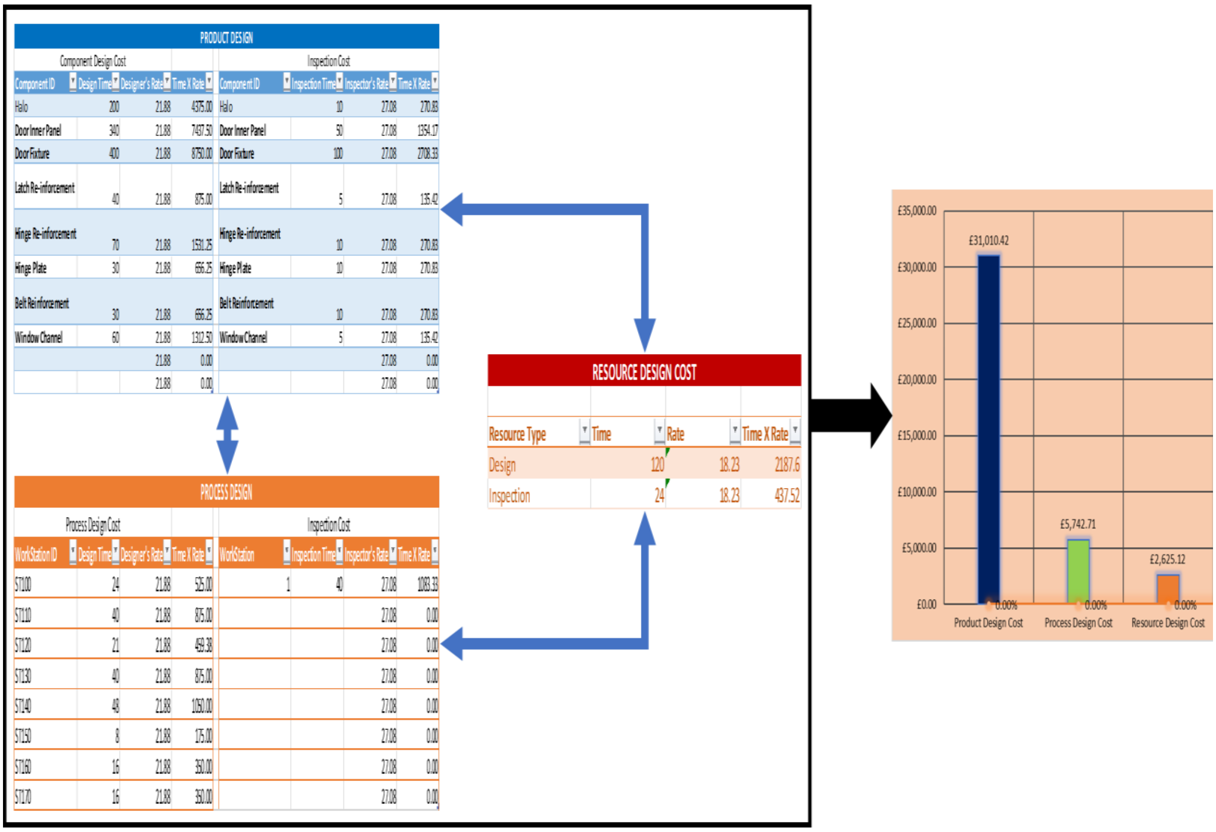


Figure 4. 17: PPR Design Cost Calculator For RLW

4.5 Feature Recognition in Cost Modeller

With the Remote Laser Welding (RLW) process, the product features considered for costing are the weld stitches. These weld stitches are expensive to make, taking into account the cost of resources required and utilized in addition to the complexity of its process design. To be able to estimate the cost involved in creating weld features at the early design stage, the cost modeller helped with product feature recognition and cost calculation make use of the customized rules, process logic and cost equations dedicated to Remote Laser Welding process within the cost modeller. The structure for identifying and recognising a product feature within the cost modeller is illustrated in Figure 4.19.

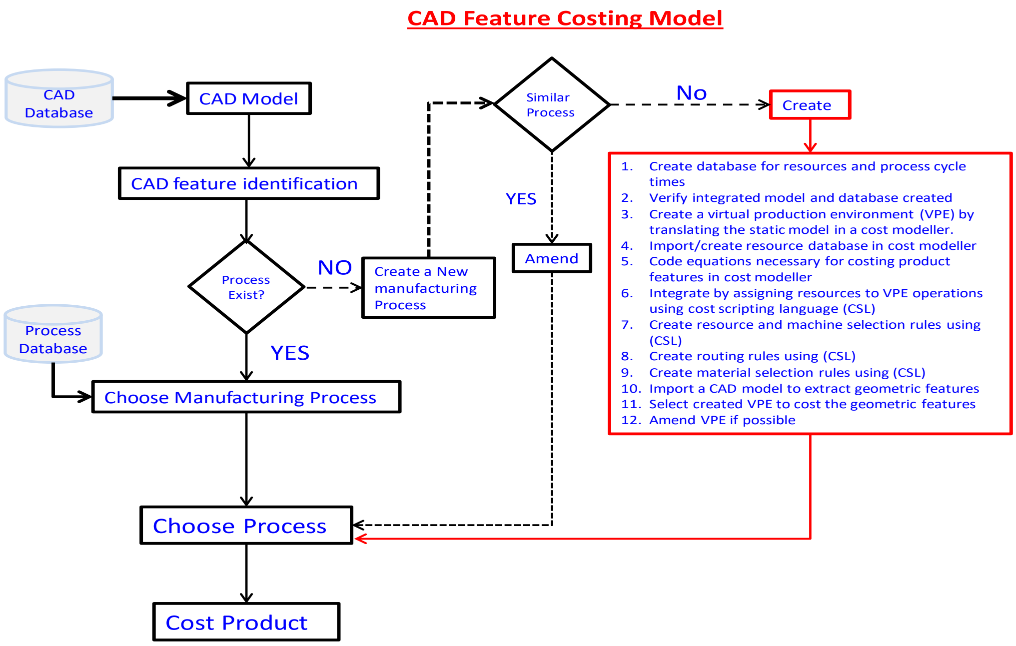


Figure 4. 18: Cost Modeller Feature Costing Principle

This section shows how the capabilities of aPriori are extended, highlighted in red in Figure 4.19 to include the Remote Laser Welding (RLW) process which is not a process option available in its database. The extension process as shown in Figure 4.19 is a contribution to this research. This required the creations of VPE and CMWB for RLW process within the Cost Engine of aPriori with limited data. RLW process in aPriori, however, assumes that the process is used for cost analysis at the design stage of a new product design where knowledge and information of product, process and resources are not fully known. The process is designed in such a way the, information and data can be added as they become available as the project progresses.

Figure 4.20 shows the cost modeller implementation structure of the PPR Cost Estimation Framework showing the flow of data and information from the Data Modelling phase to the Cost Engine Implementation phase.

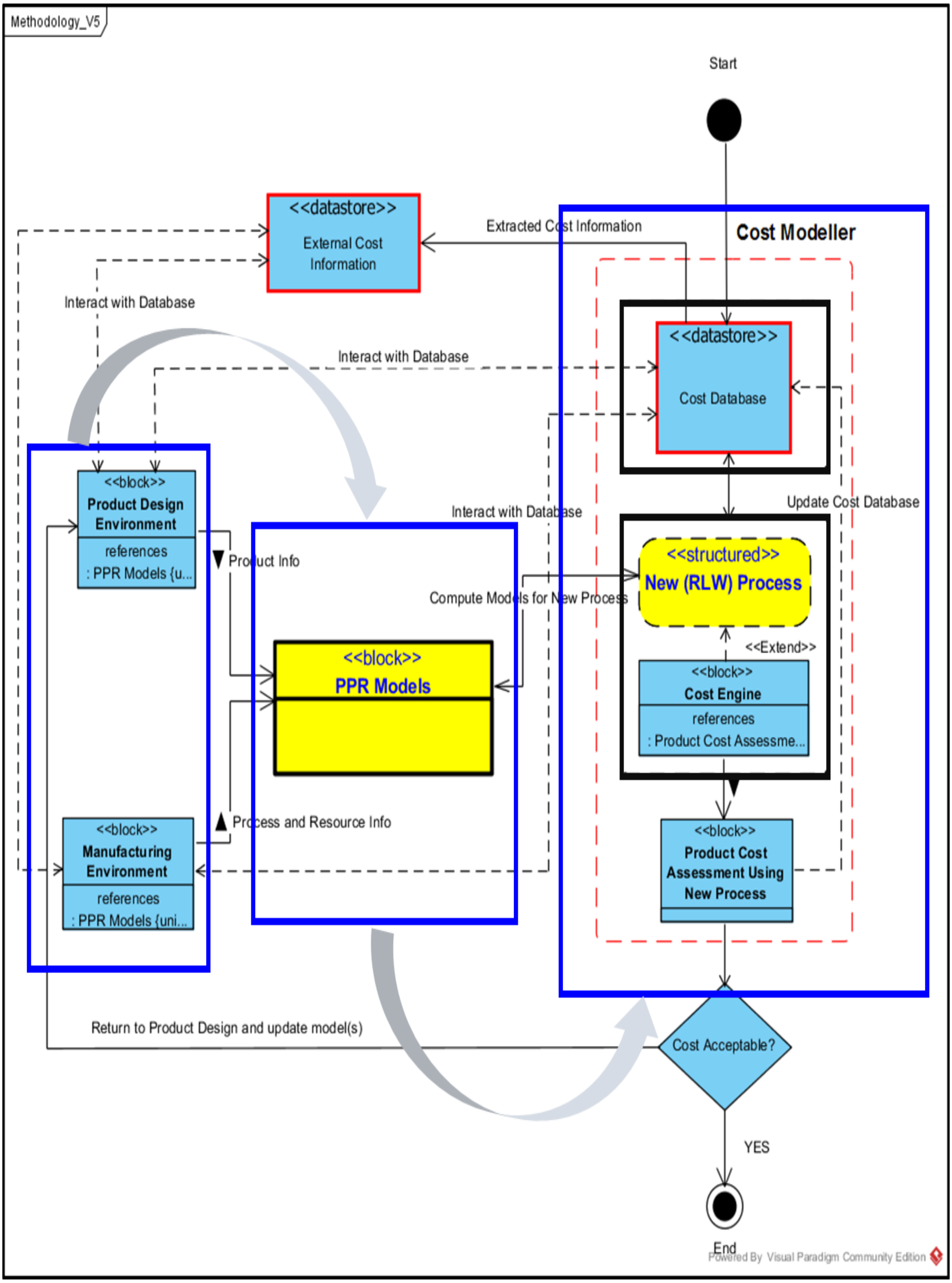


Figure 4. 19: PPR Cost Estimation Implementation

4.5.1 RLW Execution Input

Implementing the models in aPriori to represent RLW process requires the following inputs:

1. Process Sequence Script
2. Operational Sequence Script
3. Resource Sequence Script
4. Resource Database

PPR Cost Calculator - the PPR Cost Calculator was created to purposely for RLW process. The cost modeller has inbuilt cost equations which are very basic and generic with lots of assumptions. The calculator contained necessary equations based on the information available. This can be expanded as a project matures. These were coded in C++ and translated into the CSL Module of the cost engine. A snapshot of the cost equations created using CSL for the RLW process is shown in Figure 4.21.

```
 * Function : GetLaborCost
 */
GetLaborCost_Assembly_ManualMIGWelding(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_Assembly_ManualSpotWelding(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_Assembly_MechanicalAssembly(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_Assembly_PickAndPlace(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_Assembly_RoboticMIGWelding(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_Assembly_RoboticSpotWelding(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_Assembly_WeldCleanUp(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_Assembly_WeldPrep(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_Assembly_PrepareDimples(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_Assembly_RemoteLaserWelding(laborTime, cycleTime, laborRate) = LaborCost0

        LaborCost0 = GetLaborCost(laborTime, cycleTime, laborRate)
 /*
 * Function : GetDirectOverheadCost
 */
GetDirectOverheadCost_Assembly_ManualMIGWelding(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost1
GetDirectOverheadCost_Assembly_ManualSpotWelding(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost1
GetDirectOverheadCost_Assembly_MechanicalAssembly(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost1
GetDirectOverheadCost_Assembly_PickAndPlace(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost1
GetDirectOverheadCost_Assembly_RoboticMIGWelding(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost1
GetDirectOverheadCost_Assembly_RoboticSpotWelding(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost1
GetDirectOverheadCost_Assembly_WeldCleanUp(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost1
GetDirectOverheadCost_Assembly_WeldPrep(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost1
GetDirectOverheadCost_Assembly_RemoteLaserWelding(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost1
        DOHCost1 = laborCost * overheadMultiplier + overheadRate * laborTime / SEC_PER_HR
```

Figure 4. 20: A Snapshot of Costs Equations for RLK Using CSL

The complete cost equations are shown in Appendix B of this document.

4.5.2 Remote Laser Welding Implementation in aPriori's Cost Modelling Workbench.

Importing a 3D CAD model into aPriori, the weld stitch layout in the CAD model cannot be shown. In aPriori, weld can only be created by association with a reference of edges or datum curves on the 3D CAD model. Users cannot simply draw a "virtual line" to represent weld features on a product model but instead, it must be of a "solid edges" to be recognized as a feature.

As a contribution to this research, the implementation of RLW process into aPriori workbench has enable RLW stitches to be recognised by the feature recognition technology of the software. As a result, the RLW stitches is recognised by the software as a feature which has a length of 25mm with 1mm width and 1mm depth (25mm x 1mm x 1mm). Hence any feature with these characteristics on a 3D CAD model is classifies as an RLW stitch and costed using the RLW process.

The following steps were followed in developing an RLW process within the workbench of aPriori cost modeller:

Step 1: Development of Virtual Production Environment for RLW

Step 2: Create Databases and Integrate Process and Resource

Step 1: Development of Virtual Production Environment (VPE) for RLW

The virtual production environment (VPE) is a representation of the process flow mimicking the processing logic in a manufacturing environment. This takes the output of the static process model as input in the cost modeller workbench. To create the process flow for the VPE, a language compatible with the cost modeller known as cost scripting language (CSL) is used. The need for a new VPE is required because existing processes are not able to cost weld features created using remote laser welding technology.

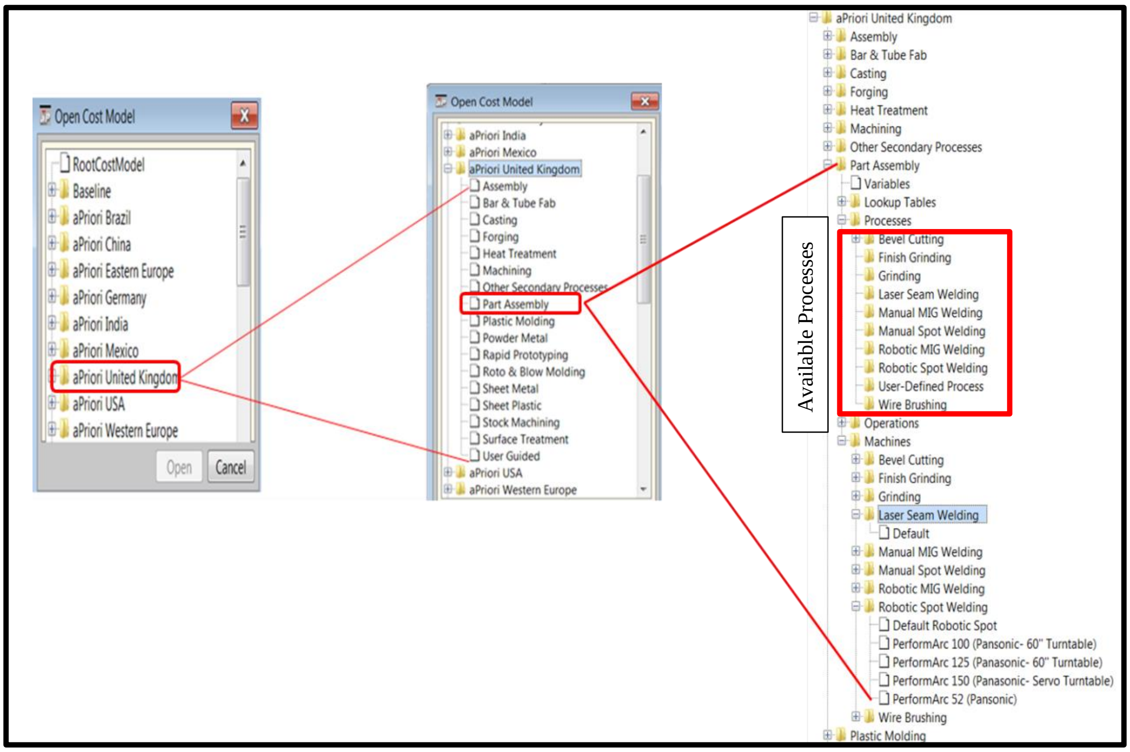


Figure 4. 21: PPR Cost Estimation Implementation

As shown in Figure 4.22, aPriori United Kingdom is the Virtual Production Environment with built-in processes available to be selected ranging from Assembly through to User-Guided. From Figure 4.22, it is obvious that Remote Laser Welding is not a process available in the cost modeller under Part Assembly as the process is designed for welding two sheet metals together. To include RLW as a process, stages 1 to 3 are to be completed as shown in Figure 4.23.

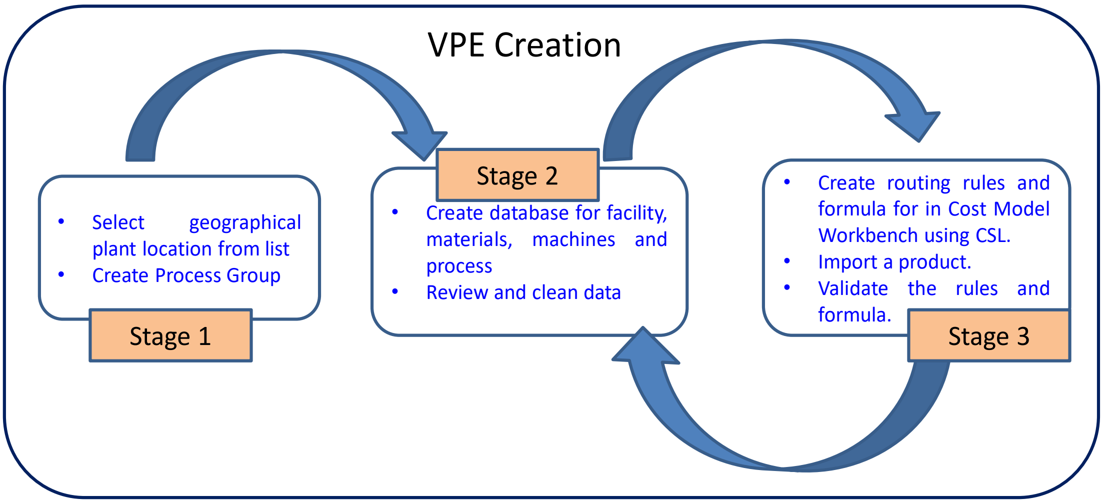


Figure 4. 22: VPE Creation Work Flow

Stage 1 - Based on the geographical location of a manufacturing plant, there is an inbuilt list of countries available for selection with predefined process groups which can also be selected or deselect for customization purposes. To create a VPE for RLW Navigator project, which is a UK based project, aPriori United Kingdom is selected as a geographical location with Assembly, Part Assembly, Sheet Metal and User-Guided Process Groups selected as shown in Figure 4.24 below.

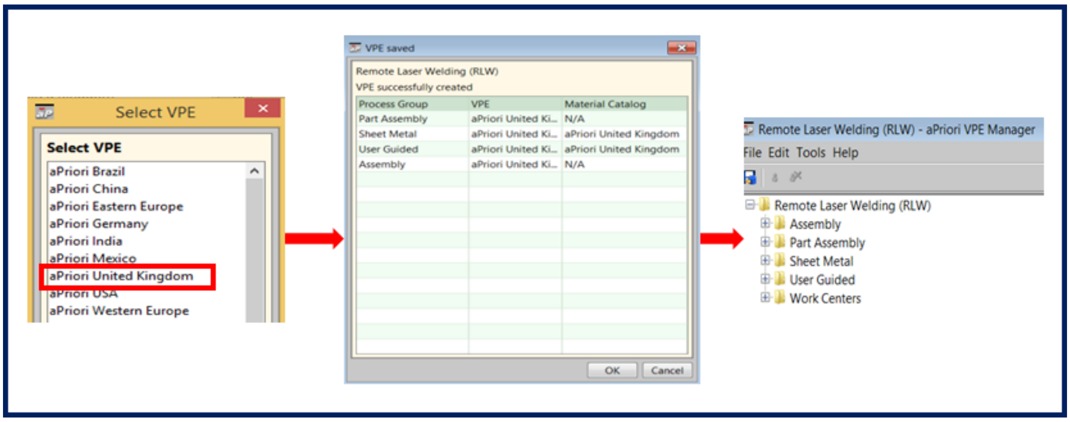


Figure 4. 23: VPE Selection and Process Group Creation

Selecting a process group for a VPE is application dependent. For this research, the case application involves welding of two sheet metals parts together which includes assembly and part assembly process sequence of sheet metals parts and also making use of user-guided functionalities of the cost modeller. Hence the purpose for which the VPE is intended for determines the process group selection as indicated in Figure 4.25.

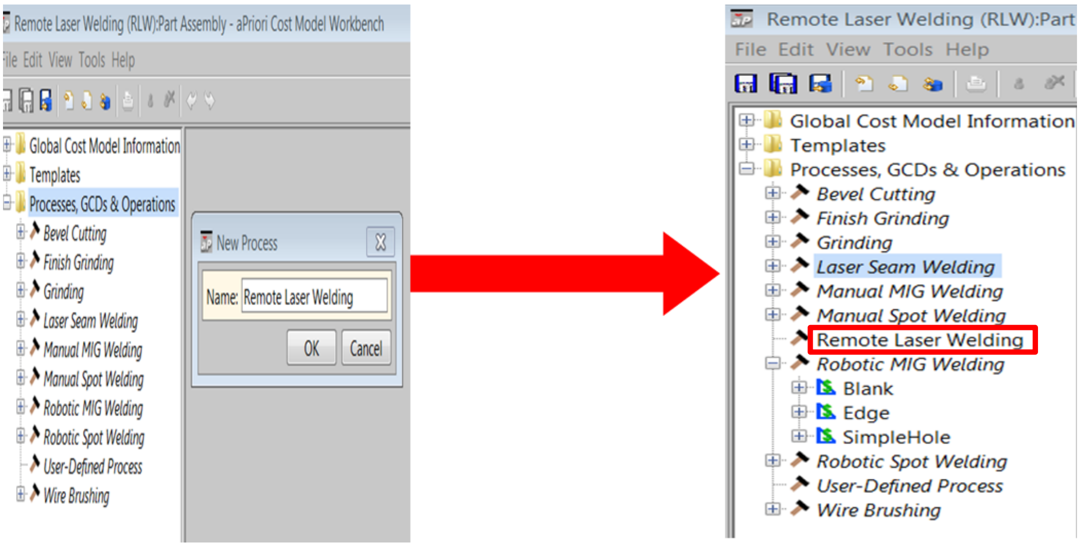


Figure 4. 24: Remote Laser Welding

This shows that Remote Laser Welding process is now created as a process option available for selection in the cost modeller. Processes such as *Laser Seam Welding, Robotic MIG Welding and Robotic Spot Welding* with "+" in front indicates that they have predefined Operations and Geometric Cost Drivers (GCDs) created for them which are necessary requirements for cost calculations. Remote Laser Welding has no Operations and Geometric Cost Drivers (GCDs) at this stage hence, not yet ready for estimating the cost of a part assembly process.

Stage 2 - Creating Operations and GCDs for Remote laser Welding process are done by creating them in Templates under the Component and Welding section. This is done by writing codes for the operations within the process using Cost Scripting Language (CSL). Input at this stage is process model created earlier, showing the sequence of operations required to realize the final welding assembly. As shown in Figure 4.25, the operation coding is done in the top section which automatically displays a graphical representation of the operational sequences. Changes to the operation can only be done in the coding section to be reflected in the template graph and not the other way round.

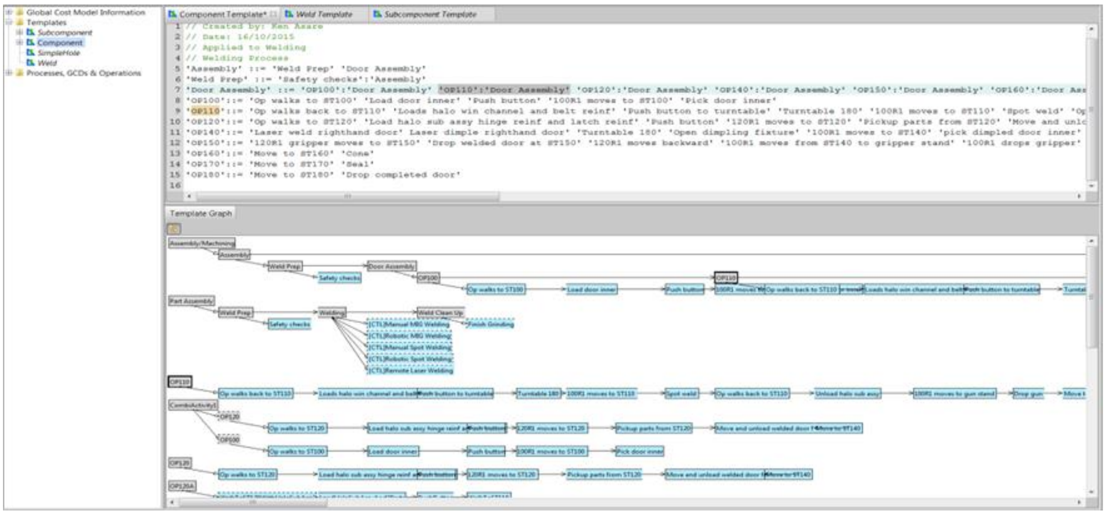


Figure 4. 25: Create Operations for Remote Laser Welding using CSL

Weld, under template also shows activities are involved specifically with the welding the welding operation. This shows how Remote Laser Welding is introduced into the cost modeller as welding process available for selection amongst other welding processes such as Manual MIG Welding, Robotic MIG Welding, Robotic Spot Welding, and Manual Spot Welding (Figure 4.26). There is no graphical display of the process sequence for Weld in the template.

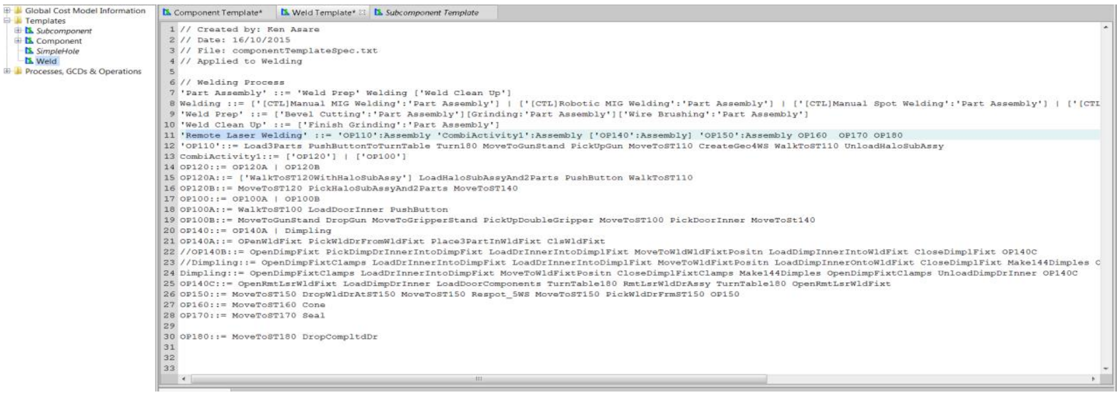


Figure 4. 26: Weld Template Creation

Once the template is set and functional, the Remote Laser Welding process database can then be created. The databases required by the cost modeller for the new process are the *CSL Modules, Process Setup Options and Machine Type*. However, *material information* is also included in this document for cost estimation.

Stage 3 (CSL Module) - Cost Scripting Language (CSL) is used for creating and implementing costing logic and rules within the Virtual Production Environment (VPE). The CSL also analysis and generates information about the part it is costing such as:

- the overall component level information and
- individual GCD, machine and material attributes.

CSL database for Remote Laser Weld process is created by expanding the green "+" symbol as shown in Figure 4.27 which gives options as to which database to create. Cost taxonomy is a required option which contains the Process Taxonomy (PTAX) and Operation Taxonomy (OTAX). PTAX rolls up calculations of child Operation Taxonomy (OTAX), whereas OTAX calculates cycle times of various operations. Database for cost taxonomy includes all cost information related to estimating the cost of the product. Figure 4.28 shows a screenshot of various information captured for the Remote Laser Welding process.

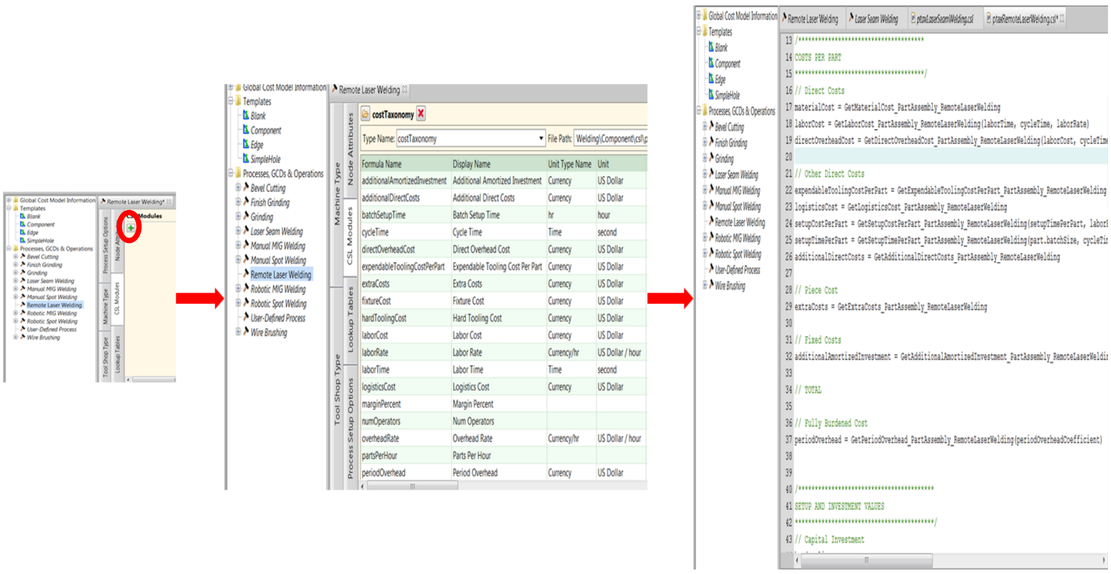
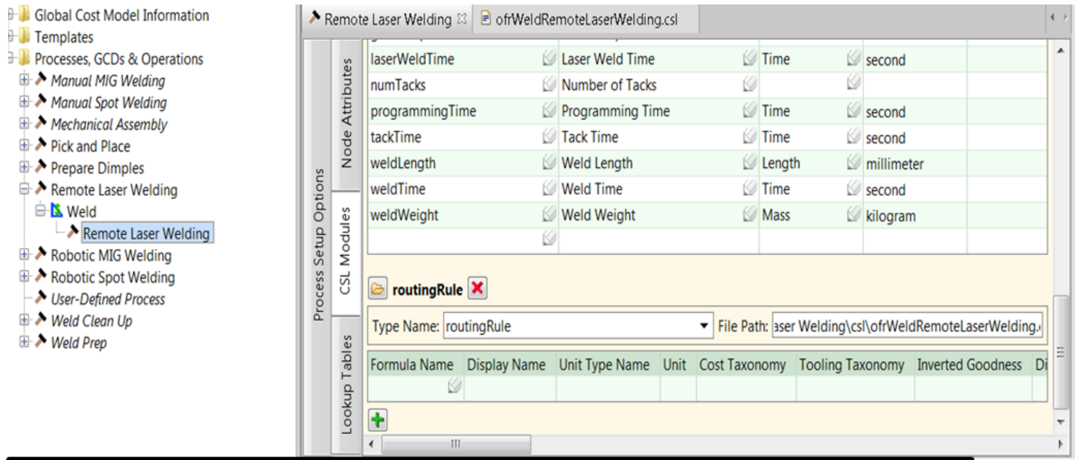


Figure 4. 27: RLW CSL Cost Taxonomy

PTAX cost definitions are coded in Process CSL cost taxonomy after capturing and entering all necessary cost information within the database. This is useful for linking all cost information associated with resources utilized and the process together. The database contains all cycle times and costs associated with the entire process.

Process Routing Rules **-** In the CSL Module, rules were created using CSL language for the process routing by specifying the welding process to be used to be Remote Laser Weld and also selecting the process as a robotic activity. This is required for the welding activity as weld stitches in the cost engine is classed as a Geometric Cost Driver. The failure message is also created to highlight malfunctions as shown in Figure 4.29.



Rule IsRemoteLaserWeld : gcd.isRobotic
Message IsRemoteLaserWeld : 'Weld type is not Robotic'

Figure 4. 28: Routing Rule Creation for RLW

Step 2*:* Create Databases and Integrate Process and Resources.

Resource database is developed in the cost modellers with information on resources populated into the database. The resource database creation is done at three levels:

Machine Type **-** Machine Type database contains information on the properties of machines needed for the process. Irrespective of the types of machine required, all their properties are stored in the Machine Type database and coding in CLS Modules makes it possible to fetch the appropriate property needed for each machine. Parameters such cost, time and ratios of resources are captured at this stage in the database with their descriptions and value. Although this is an optional part of the cost modeller, information entered affects result when costing. Therefore, as resource information becomes available, they are added to generate an up to dated results. Once this is done, resources needed for the Remote Laser Welding process are created with details of their costs, rates and time information as shown in Figure 4.30.

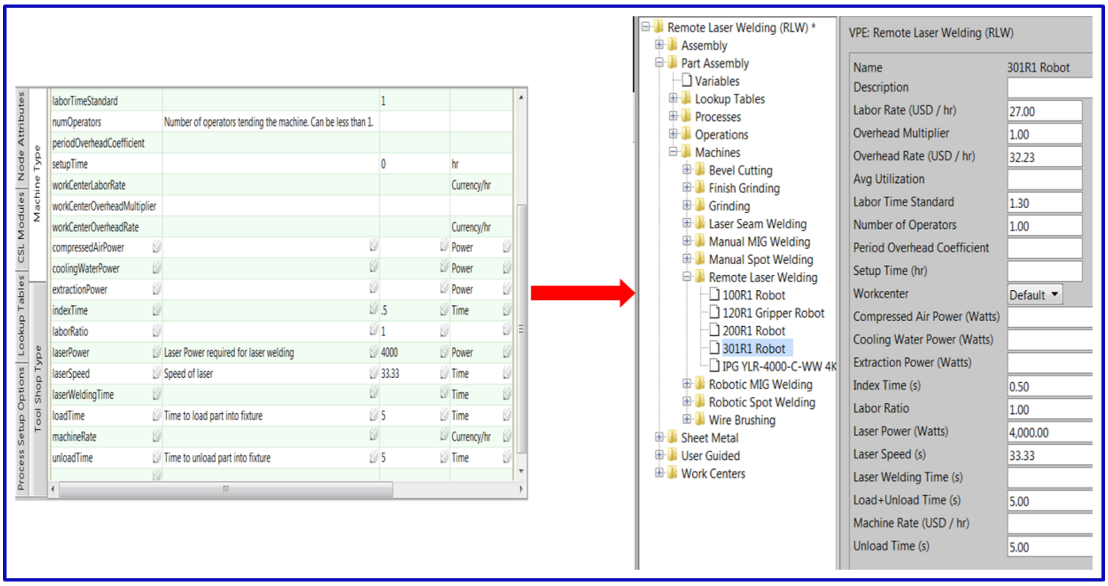


Figure 4. 29: RLW Machine Type and Information

Also, Figure 4.30 shows the types of machine resources (robots: *100R1 Robot, 200R1 Gripper Robot, 200R1 Robot and IPG YLR-400-c-WW 4K*) under the Remote Laser Welding process with available resource imputation. Other parameters may be added to Machine Type database for resources as they become available. Once the machine type data are entered, rules are then created to integrate and assign resources to RLW operations. This is done by coding using cost scripting language (CSL) language. The code is then implemented in the cost engine for machine selection as well as displaying failure messages where machines are not capable of realizing tasks.

***Process Setup Options* -** The Process Setup Options (PSO) in the cost engine is used for entering values and description for activities such as; number of operators, fixture loading and unloading times and cycle times needed for completing the welding process. Values entered in PSO affects the overall cost of the welding process as shown in Figure 4.31. This works with codes written in the CLS module, hence inputs must be the same for both sections to be able to cost using the VPE created for the RLW process.

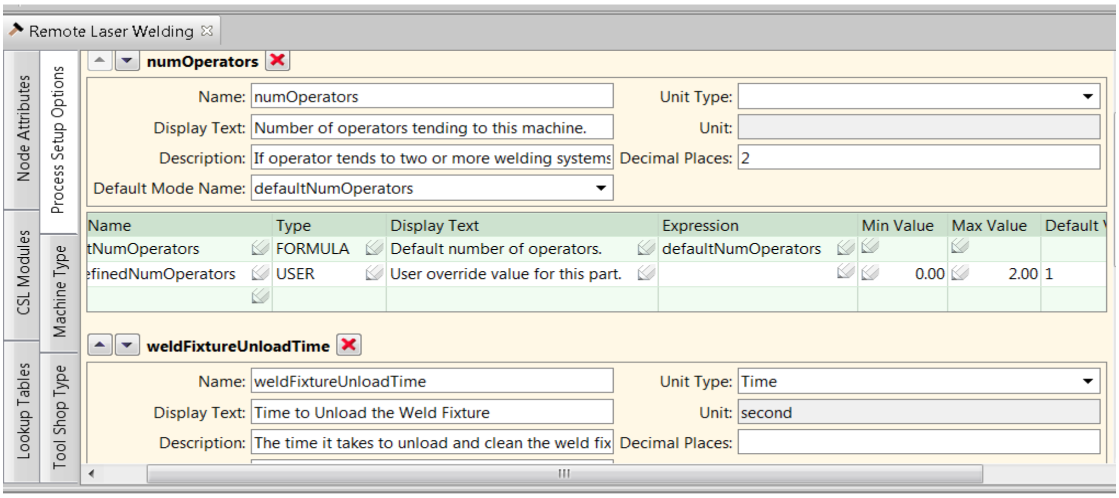


Figure 4. 30: RLW Process Setup Options

Cost Equations – Equations created in the cost engine for calculating the cost of the assembly process using CSL are shown in Figure 4.32. The complete cost equations are displayed in Appendix B.

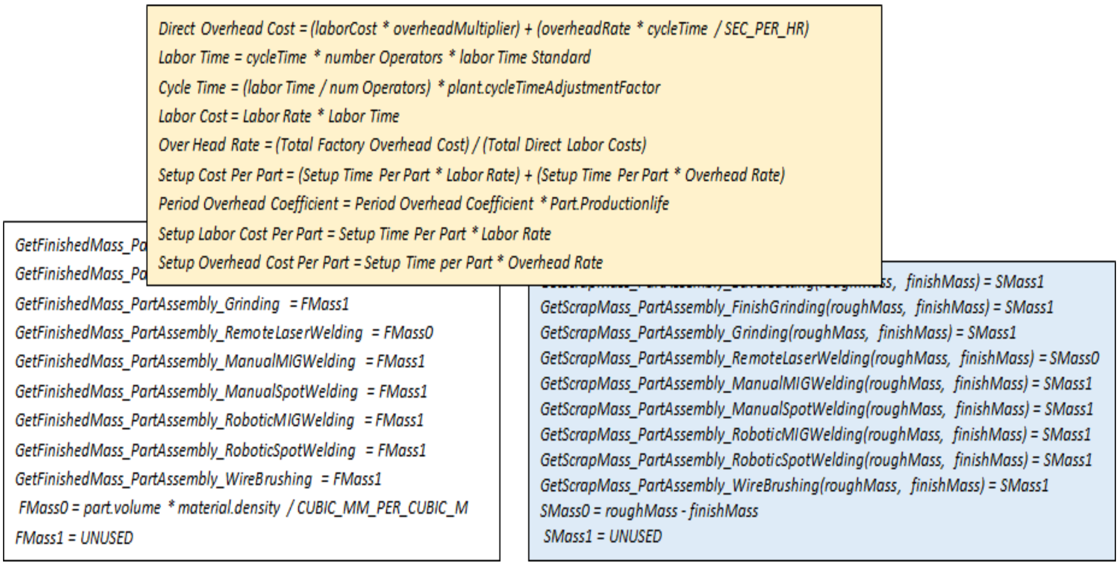


Figure 4. 31: RLW Cost Equations

Material Data: Material data contain information and properties of the material that could potentially be used to realize the product. Available material information is updated and customised material data (Galvanised steel as material types with item names DX52+Z, DX53+Z, DX54+Z and DX56+Z) specifically for the Remote Laser Welding process is created and added to existing material data as shown in Table 4.5.

these contain unit cost per kg information as this material type is preferred to be used in the automotive industry for the body in white (BIW) applications.

Table 4. 5: Material Data

	A	B	C	D	E	F	G	H	I	J
1	processGroupName	Name	Description	Material Type	Cut Code	USA Name	DIN Name	EN Name	Unit Cost (USD / kg)	Cost Units
2	Sheet Metal	10025-2 CR	A109, A576	Steel	1.1	Steel- CR- 1012	1.0144 CR	10025-2 CR	0.97636694	costPerKG
3	Sheet Metal	10025-2 HR	A108, A635	Steel	1.1	Steel- HR- 1012	1.0144 HR	10025-2 HR	0.866708959	costPerKG
4	Sheet Metal	10025/2-2004 HR	A108, A519, A569, A5	Galvanized Steel	1.1	Galv. Steel- HR- 1010	59231 HR	10025/2-2004 HR	1.071322221	costPerKG
5	Sheet Metal	10028-7	A167, A240, A276, A6	Stainless Steel	5.21	Stainless- 304	1.4301	10028-7	3.235688271	costPerKG
6	Sheet Metal	10083-2 CR	A109, A635, A659	Steel	1.1	Steel- CR- 1020	1.0402 CR	10083-2 CR	0.986845945	costPerKG
7	Sheet Metal	10083-2 HR	A108, A635, A659	Steel	1.1	Steel- HR- 1020	1.0402 HR	10083-2 HR	0.874806004	costPerKG
8	Sheet Metal	10088-2	A167, A240, A276	Stainless Steel	5.21	Stainless- 316L	1.4404	10088-2	4.496719471	costPerKG
9	Sheet Metal	10088-7	A167, A240, A260, A2	Stainless Steel	5.21	Stainless- 316	17441	10088-7	4.472468871	costPerKG
10	Sheet Metal	10132-2 CR	A109, A576	Steel	1.1	Steel- CR- 1010	1.1121 CR	10132-2 CR	0.937943921	costPerKG
11	Sheet Metal	10132-2 HR	A108, A635	Steel	1.1	Steel- HR- 1010	1.1121 HR	10132-2 HR	0.822753571	costPerKG
12	Sheet Metal	10210 HR	A108, A569, A570, A6	Galvanized Steel	1.1	Galv. Steel- HR- 1020	1626 HR	10210 HR	1.184851052	costPerKG
13	Sheet Metal	10277-2 HR	A108, A569, A570, A6	Galvanized Steel	1.1	Galv. Steel- HR- 1012	10142 HR	10277-2 HR	1.17184254	costPerKG
14	Sheet Metal	4004	B209, B211	Aluminum	30.11	Aluminum- 3003	3.0517	4004	6.575458632	costPerKG
15	Sheet Metal	4004	B209, B211	Aluminum	30.11	Aluminum- 3003	3.0517	4004	6.575458632	costPerKG
16	Sheet Metal	485-2	B209, B221	Aluminum	30.11	Aluminum- 6061	3.3211	485-2	3.764537632	costPerKG
17	Sheet Metal	AW5052	B209, B211	Aluminum	30.11	Aluminum- 5052	3.3523	AW5052	2.778858231	costPerKG
18	Sheet Metal	Cu-ETP	B36, B152, B370	Copper	33.3	Copper- 110	1787	Cu-ETP	8.690953047	costPerKG
19	Sheet Metal	DX52 + Z	DX52 + Z	Galvanized Steel	1.1	Galv. Steel- 1.0350	59231 HR	DX52 + Z	1.184851	costPerKG
20	Sheet Metal	DX53 + Z	DX53 + Z	Galvanized Steel	1.1	Galv. Steel- 1.0351	59231 HR	DX53 + Z	1.184851	costPerKG
21	Sheet Metal	DX54 + Z	DX54 + Z	Galvanized Steel	1.1	Galv. Steel- 1.0352	59231 HR	DX54 + Z	1.171843	costPerKG
22	Sheet Metal	DX56 + Z	DX56 + Z	Galvanized Steel	1.1	Galv. Steel- 1.0353	59231 HR	DX56 + Z	1.171843	costPerKG

The total cost of material for the process is the sum of all material costs for individual components in the assembly. Figure 4.33 shows a snapshot of some of the cost definitions. The formula coded in the cost engine using the Cost Scripting Language (CSL) to calculate the cost of sheet metal components of the car door is:

*Material Cost = (Material Cost Per Mass * Part Mass) / Utilization*

Where;

Part Mass = Part Volume x Material Density
Material cost per mass = Material cost per each kilogram
Utilization = Rough Mass / Finish Mass
Scrap Mass = Rough Mass - Finish Mass
Rough Mass - The initial part mass, kg
Finish Mass - The final part mass, kg

Figure 4. 32: Material Cost Definition

Once CSL modules, Machine Type, Process Setup Options and Material data are created for Remote Laser Welding process, the Virtual Production Environment (VPE) is then ready to be used for costing welding operation for RLW technology.

4.6 Cost Assessment stage for RLW

The cost assessment is carried out for the application of the RLW technology for making weld stitches on the door assembly. This is done in the following categories:

1. For assessing the cost of making 70 to 75 weld stitches of 25mm length in batches of 10000, 20000, 30000, 40000, 50000, 60000, 70000, 80000, 90000, 100000, 500000 and 1000000 doors. This assumes the useful life of the manufacturing environment to be 10 years using the newly created manufacturing process.
2. Assessing the cost of individual components using alternative materials (DX52+Z, DX53+Z, DX54+Z and DX56+Z) and

To achieve these, the following steps must be followed:

***CAD Model Import* -** To validate the VPE created for the Remote Laser Welding (RLW) technology, a CAD model is imported and the VPE created is applied to cost the CAD. As mentioned earlier, all VPEs contain predefined resources and those resources cannot be changed or amended. When using an existing VPE, the cost engine assumes that all resources are 100 percent available at all times but in practice, this is not true. That means that existing VPEs are not capable of costing the remote laser welding stitches, hence, the newly developed virtual production environment (VPE) has to be selected for assessing the cost of making weld stitches on the car doors. The cost value outputs obtained from aPrior for the RLW process was validated by comparing the values with manually calculated cost using traditional cost accounting principles which were also verified to be right by the RLW project's industrial experts. The VPE created is dedicated to costing virtual weld stitches with dedicated resources with process and operations cycle times representing a typical manufacturing environment.

A CAD model of an assembled car door L358 (FDRBJ32220124A01V2) was imported into the cost engine for cost calculation and analysis. Initial importation of a CAD model uses the last used VPE for cost calculation. Selecting the VPE created for Remote Laser Welding from the Manufacturing Process column in Figure 4.34 can be done VPE from the list of stored VPEs. Year 3 VPE is now available for costing the assembly process, showing machine details as created in the machine database.

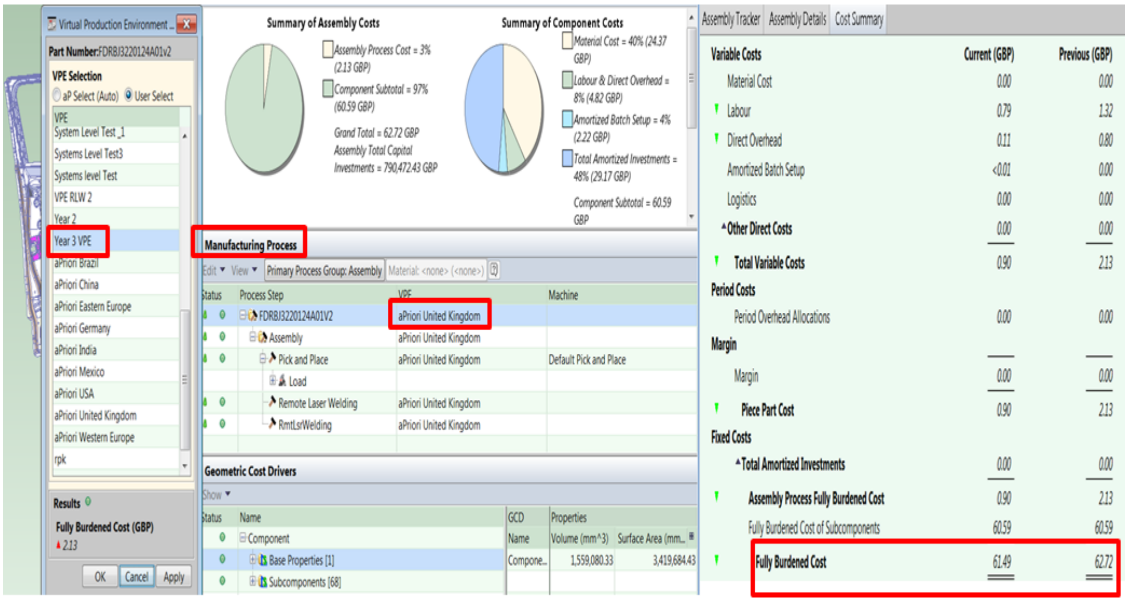


Figure 4. 33: CAD Model Import

The Cost Summary gives a breakdown of the costs associated with the VPE selected and as shown, Material Cost is £0.00 because the VPE is designed as an Assembly Process hence, material cost is added later for a complete cost calculation. Also, the cost of Remote Laser Welding stitches are not included in this initial CAD model cost. The Fully Burdened Cost of the assembly is the cost of the sum of operations including pick and place, loading, unloading and clamping activities and the cost of resources assigned to the manufacturing process.

RLW Stitches Feature Definition - As RLW process and resources are not available within the cost modeller, recognising laser welding feature and costing them was not possible (See Figure 4.35). To overcome this, weld stitches locations were defined on the product CAD model to be welded. Stitch allocations are also used other optimization purposes such as clamp location and robot trajectory.

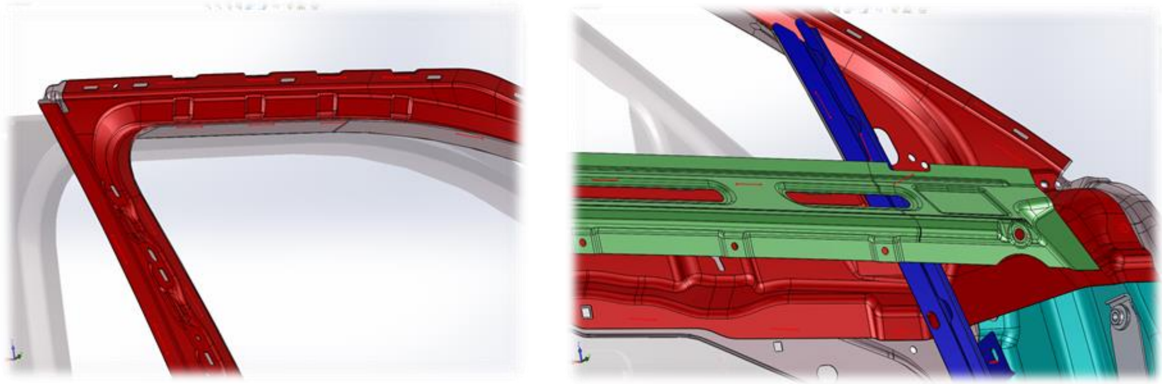


Figure 4. 34: Laser Stitch Problem

The feature recognition technology inbuilt in the cost modeller is trained to identify marks made on CAD model of 25mm length with 1mm thickness as a remote laser weld stitch as shown in Figure 4.36.

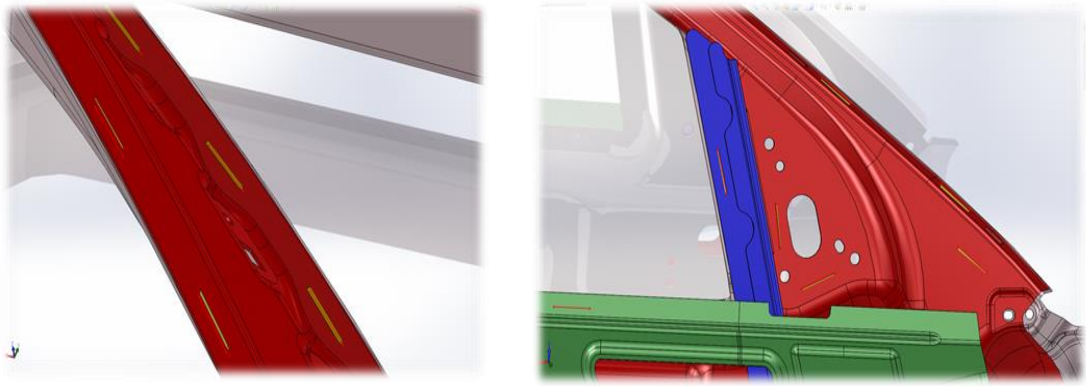


Figure 4. 35: Laser Weld Stitches Solution

Weld length, thickness and weld type are all predefined for the welding task. Using the Year 3 VPE created, cost of welding is added to the initial cost information as the number of stitches increases. This is demonstrated in Figure 4.36 with weld length and pitch set at 25mm and 75mm respectively for all 72 stitches.

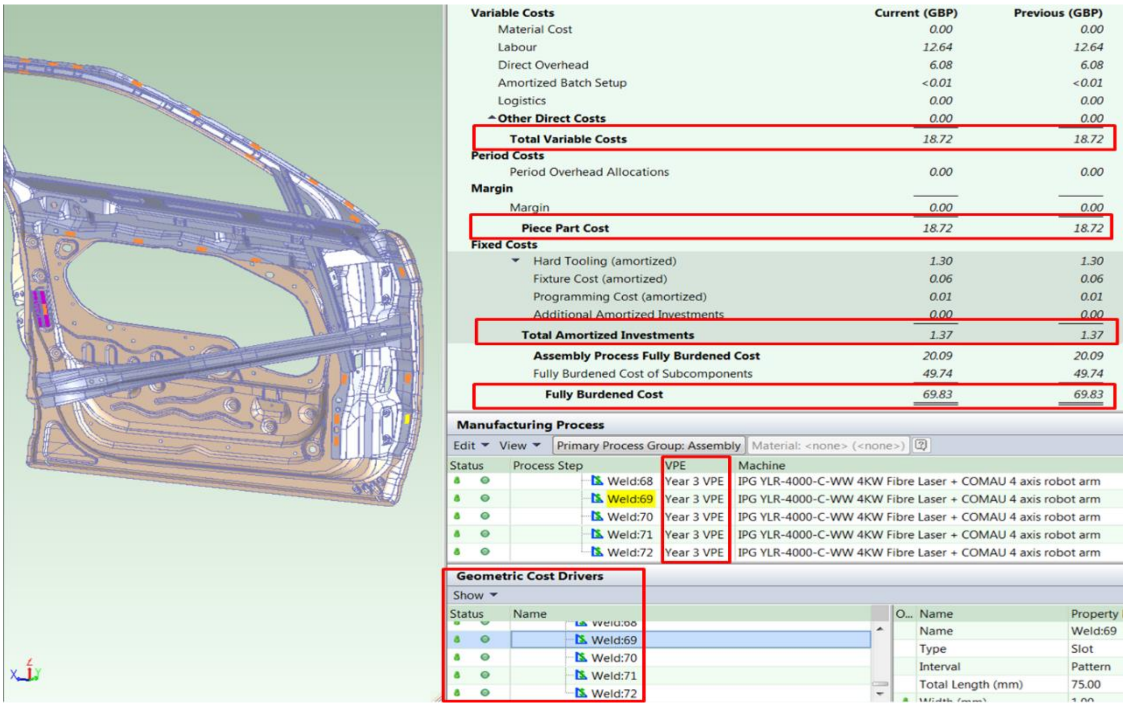
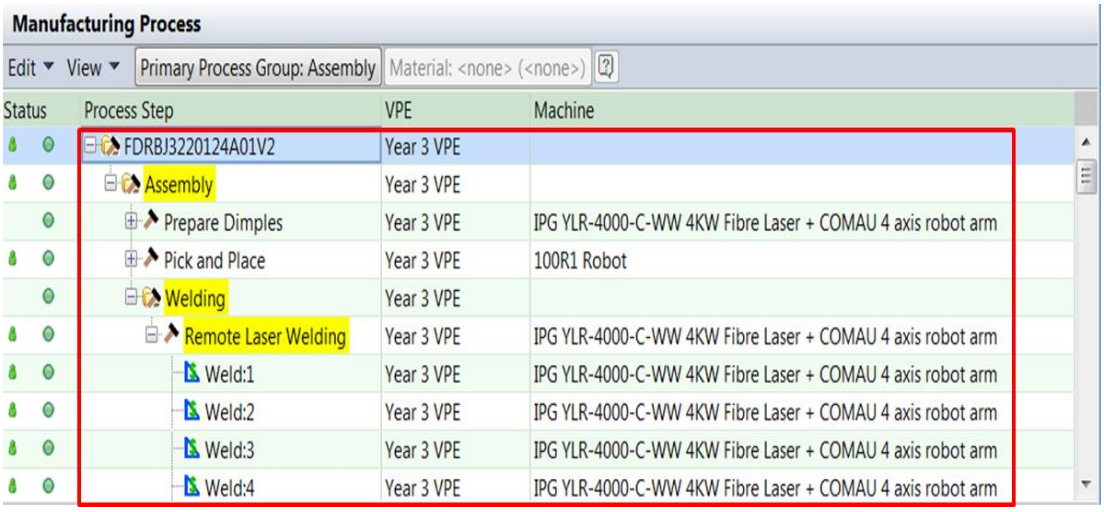


Figure 4. 36: RLW Weld Stitches Creation

The Geometric Cost Driver (GCD) identifies the weld stitches as features on the CAD model, therefore as the weld stitches increases, Fully Burdened Cost also increases as shown in Figure 4.37. Also, in the Manufacturing Process section, under VPE, the VPE is changed to Year 3 VPE specifically designed for Remote Laser Welding process. Under machine, IPG YLR-4000-C-WW 4KW Fibre Laser + COMAU 4 axis robot arm is selected and used for preparing Dimple and Remote Laser Welding tasks. For Pick and Place operations, 100R1 Robot which is a grabbing robot is used.



Manufacturing Process

Edit ▾ View ▾ | Primary Process Group: Assembly | Material: <none> (<none>)

Status	Process Step	VPE	Machine
	FDRBJ3220124A01V2	Year 3 VPE	
	Assembly	Year 3 VPE	
	Prepare Dimples	Year 3 VPE	IPG YLR-4000-C-WW 4KW Fibre Laser + COMAU 4 axis robot arm
	Pick and Place	Year 3 VPE	100R1 Robot
	Welding	Year 3 VPE	
	Remote Laser Welding	Year 3 VPE	IPG YLR-4000-C-WW 4KW Fibre Laser + COMAU 4 axis robot arm
	Weld:1	Year 3 VPE	IPG YLR-4000-C-WW 4KW Fibre Laser + COMAU 4 axis robot arm
	Weld:2	Year 3 VPE	IPG YLR-4000-C-WW 4KW Fibre Laser + COMAU 4 axis robot arm
	Weld:3	Year 3 VPE	IPG YLR-4000-C-WW 4KW Fibre Laser + COMAU 4 axis robot arm
	Weld:4	Year 3 VPE	IPG YLR-4000-C-WW 4KW Fibre Laser + COMAU 4 axis robot arm

Figure 4. 37: VPE and Machines Selection

The Status column reports any problems associated with the VPE or the machine selection but Figure 4.38 shows all the green dots and green dude indicating that there is no error with the VPE. Therefore, the cost engine is capable of calculating Remote Laser Welding process cost in a manufacturing sector to support engineering decision making.

Routing Selection **-** Under Manufacturing Process in aPriori, Remote Laser Welding process can then be selected as shown in Figure 4.39. All other welding processes are excluded, leaving Remote Laser Welding process as the only process for costing the assembly process. This is because RLW process is integrated with material database specifically designed for the process and therefore excluded processes are indicated with a red 'X' attached to them.

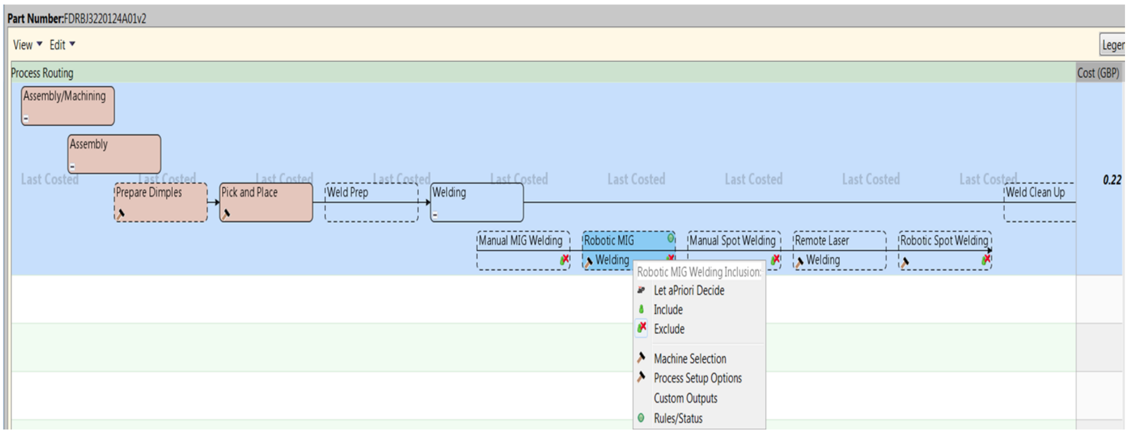


Figure 4. 38: Routing Selection

4.6.1 RLW Installation Cost Calculation

Using equations (3.22) to (3.28) from Chapter 3, the installation cost for the RLW process is the sum of mechanical, electrical and IT installation costs.

$$C_{instal\_RLW} = \sum (C_{mec\_RLW} + C_{elect\_RLW} + C_{it\_RLW})$$

$$C_{mec_{RLW}} = Mechanical\ Engineer's\ Rate \times Setup\ Time$$

$$= £30/hr \times 480hr$$

$$= £14400$$

$$C_{elect_{RLW}} = Electrical\ Engineer's\ Rate \times Setup\ Time$$

$$= £30/hr \times 160hr$$

$$= £4800$$

$$C_{it_{RLW}} = IT\ Engineer's\ Rate \times Setup\ Time$$

$$= £35/hr \times 160hr$$

$$= £5600$$

Therefore, Total RLW installation cost

$$C_{instal\_RLW} = \sum (14400 + 4800 + 5600)$$

$$= £24800$$

RLW Process and Resource Integration Output: Outputs for the RLW process and resource integration are:

- RLW workstations and operations integration diagram (Figures 3.18)
- RLW workstations and operations integration script (Figures 3.19) and finally

- the RLW Integrated Cost Equations at the workstations level (Table 3.25)

4.7 Results for RLW Process

This section shows the results of the RLW case application using the PPR Cost Estimation Framework. These results consist of product design cost, process design cost, production cost and installation cost.

4.7.1 Product Design Cost

Total Product Design Cost was calculated using equation (3.1), which is the sum of all the RLW components design costs. Table 4.6 shows the breakdown of the design cost of the RLW product. Design cost and Inspection costs are separated to help understand the cost components.

Table 4. 6: Product Design Cost

PRODUCT DESIGN COST										
Component Design Cost							Inspection Cost			
Component Name	Design Time	Rework Time	Designer's Rate	Rework Cost	Design Cost		Component	Inspection Time	Inspector's Rate	Cost
Halo	200	5	19.27	96.35	3854.17		Halo	10	26.04	260.42
Door Inner Panel	340	6	19.27	115.63	6552.08		Door Inner Panel	50	26.04	1302.08
Door Fixture	400	12	19.27	231.25	7708.33		Door Fixture	100	26.04	2604.17
Latch Re-inforcement	40	4	19.27	77.08	770.83		Latch Re-inforcement	5	26.04	130.21
Hinge Re-inforcement	70	3	19.27	57.81	1348.96		Hinge Re-inforcement	10	26.04	260.42
Hinge Plate	30	3	19.27	57.81	578.13		Hinge Plate	10	26.04	260.42
Belt Reinforcement	30	4	19.27	77.08	578.13		Belt Reinforcement	10	26.04	260.42
Window Channel	60	4	19.27	77.08	1156.25		Window Channel	5	26.04	130.21
			19.27	0.00	0.00				26.04	0.00

4.7.2 Process Design Cost

The process design cost results are shown in Table 4.7, showing the cost components of the RLW process and the parameters used to calculate the process design cost.

Table 4. 7: Process Design Cost Results

PROCESS DESIGN COST									
Process Design Cost						Inspection Cost			
WorkStation Name	Design Time	Rework Time	Designer's Rate	Rework Cost	Design Cost	WorkStation	Inspection Time	Inspector's Rate	Cost
ST100	10	2	27.60	55.21	276.04	ST100	2	31.25	62.50
ST110	40	4	27.60	110.42	1104.17	ST110	4	31.25	125.00
ST120	8	4	27.60	110.42	220.83	ST120	2	31.25	62.50
ST130	48	4	27.60	110.42	1325.00	ST130	4	31.25	125.00
ST140	15	3	27.60	82.81	414.06	ST140	5	31.25	156.25
ST150	4	3	27.60	82.81	110.42	ST150	1	31.25	31.25
ST160	4	3	27.60	82.81	110.42	ST160	1	31.25	31.25
ST170	5	3	27.60	82.81	138.02	ST170	2	31.25	62.50
			27.60	0.00	0.00			31.25	0.00

4.7.3 Resource Design Cost

The cost of the resource design is the cost associated with designing and modelling all the resources required for the RLW process. Table 4.8 shows the cost summary of the resource design cost.

Table 4. 8: Resource Design Cost

RESOURCE DESIGN COST									
Resource Design Cost						Inspection Cost			
Resource Name	Design Time	Rework Time	Designer's Rate	Rework Cost	Design Cost	WorkStation	Inspection Time	Inspector's Rate	Cost
100R1	80	5	23.44	117.1875	1875	100R1	2	29.69	59.38
120R1	50		23.44	0	1171.875	120R1	4	29.69	118.75
130R1	40		23.44	0	937.5	130R1	2	29.69	59.38
140R1	160		23.44	0	3750	140R1	4	29.69	118.75
Turntable 1	30		23.44	0	703.125	Turntable 1	5	29.69	148.44
Turntable 2	22		23.44	0	515.625	Turntable 2	1	29.69	29.69
Dimpling Fixture	60	10	23.44	234.375	1406.25	Dimpling Fixture	1	29.69	29.69
Welding Fixture	200	15	23.44	351.5625	4687.5	Welding Fixture	30	29.69	890.63
Putdown Fixture	20		23.44	0	468.75	Putdown Fixture	4	29.69	118.75
Coning Machine	10		23.44	0	234.375	Coning Machine	2	29.69	59.38
Sealing Machine	10		23.44	0	234.375	Sealing Machine	2	29.69	59.38
			23.44	0	0			29.69	0.00

The total cost of designing the RLW resources is the sum of resource design and inspection costs, which is £15,984.375.

4.7.4 Installation Cost

The summary of the cost of setting up and installing the RLW process is shown in Table 4.10 indicating each cost components with their cost values. Same procedures used for calculating PPR cost were used to calculate the installation costs. Therefore, the total cost of installation is £12880 for the RLW Navigator project.

Table 4. 9: Installation Cost Results

INSTALLATION COST			
Engineer	Rate	Time	Time X Rate
Mechanical	25	240	6000
Electrical	28	160	4480
IT	30	80	2400

4.7.5 Remote Laser Welding Process Cost

Several virtual experiments were carried out using the Taguchi design of experiment (DoE) to analyse the manufacturing cost of using the remote laser welding process to make weld stitches on the door. The experimental design consists of varying manufacturing parameters such as annual volume, batch size and number of weld stitches. However, other manufacturing cost parameters such as production life which was set to 10 years and material type used was DX52+Z were unchanged throughout the virtual experiments.

The annual volumes were set at 10000, 20000, 30000, through to 100000 for the experiment having an interval of 10000 volumes per annum. Batch sizes selected were 1000, 2000, 3000, through to 10000, having an interval of 1000. Also, the number of weld stitches were also varied from 70 stitches to 75 stitches where each stitch is assumed to have a length of 25mm, a thickness of 1mm and a width of 1mm as shown in Figure 4.40.

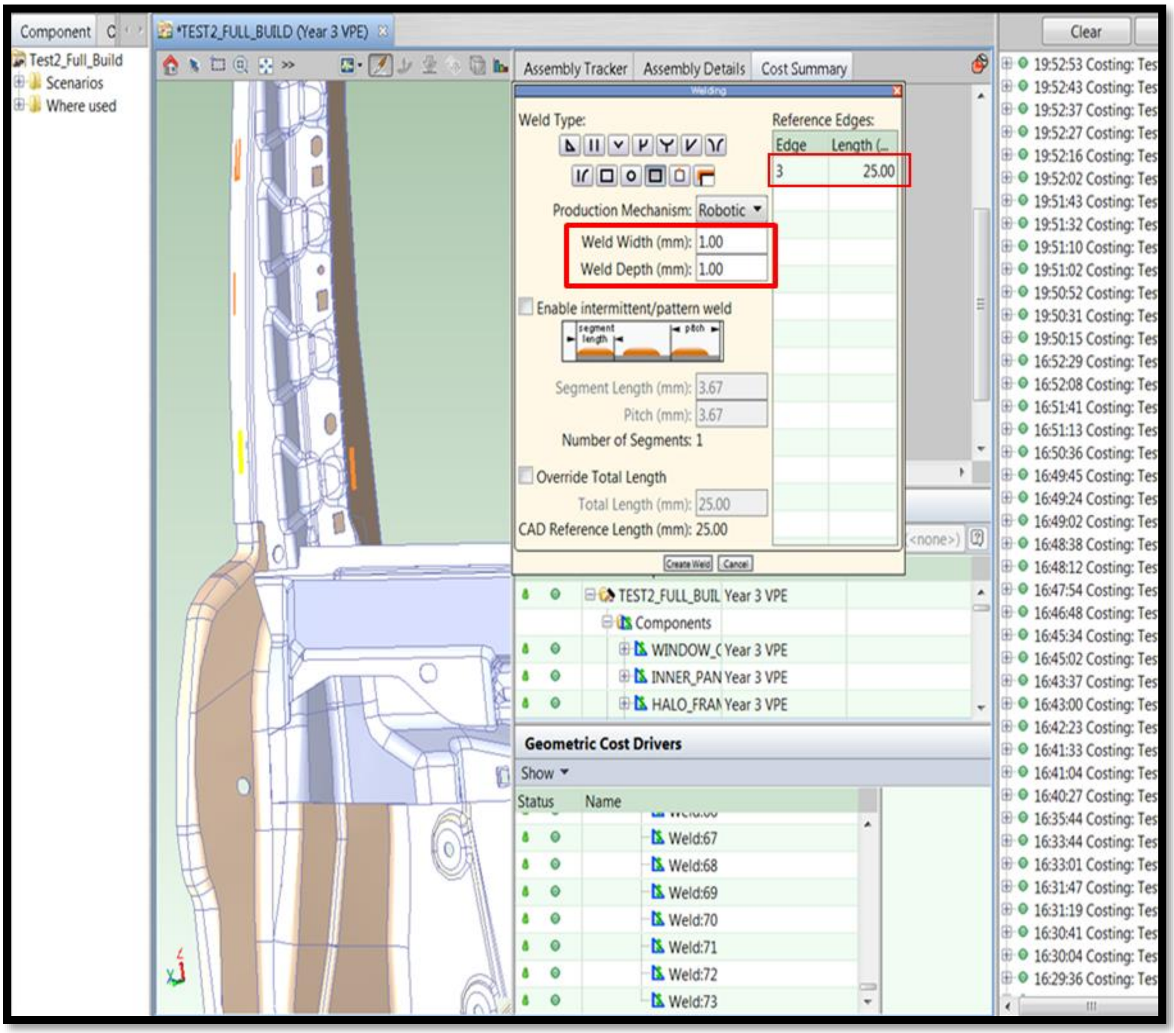


Figure 4. 39: Experimental Parameters in aPriori

Table 4.11 shows the Taguchi's design of experiment used for the welding process cost. 54 experiments were conducted for 70, 71,72, 73, 74 and 75 weld stitches on the door. For each number of weld stitches, annual volumes of 10000, 50000 and 100000 were used with batch sizes of 1000, 5000 and 10000. The same sub component cost was applied because one material type was used for the door in this experiment.

Table 4. 10: RLW Process Results using Design of Experiment

Number of Stitch	Annual Volume	Batch Size	Pattern	RLW Fully Burdened Cost (£)	RLW Fully Burdened Cost Breakdown (£)		
					Assembly Process	Sub Component	Annual
70	10000	1000	−−−	30.42	11.56	18.86	304,200.00
70	10000	5000	−−0	30.41	11.55	18.86	304,100.00
70	10000	10000	−−+	30.41	11.55	18.86	304,100.00
70	50000	1000	−0−	24.95	6.08	18.86	1,247,500.00
70	50000	5000	−00	24.94	6.07	18.86	1,247,000.00
70	50000	10000	−0+	24.94	6.07	18.86	1,247,000.00
70	100000	1000	−+−	24.26	5.40	18.86	2,426,000.00
70	100000	5000	−+0	24.25	5.39	18.86	2,425,000.00
70	100000	10000	−++	24.25	5.39	18.86	2,425,000.00
71	10000	1000	−−−	30.47	11.61	18.86	304,700.00
71	10000	5000	−−0	30.46	11.60	18.86	304,600.00
71	10000	10000	−−+	30.46	11.60	18.86	304,600.00
71	50000	1000	−0−	25.00	6.13	18.86	1,250,000.00
71	50000	5000	−00	24.99	6.12	18.86	1,249,500.00
71	50000	10000	−0+	24.98	6.12	18.86	1,249,000.00
71	100000	1000	−+−	24.31	5.45	18.86	2,431,000.00
71	100000	5000	−+0	24.30	5.44	18.86	2,430,000.00
71	100000	10000	−++	24.30	5.44	18.86	2,430,000.00
72	10000	1000	0−−	30.52	11.66	18.86	305,200.00
72	10000	5000	0−0	30.51	11.65	18.86	305,100.00
72	10000	10000	0−+	30.51	11.65	18.86	305,100.00
72	50000	1000	00−	25.04	6.18	18.86	1,252,000.00
72	50000	5000	0	25.03	6.17	18.86	1,251,500.00
72	50000	10000	00+	25.03	6.17	18.86	1,251,500.00
72	100000	1000	0+−	24.36	5.50	18.86	2,436,000.00
72	100000	5000	0+0	24.35	5.49	18.86	2,435,000.00
72	100000	10000	0++	24.35	5.49	18.86	2,435,000.00
73	10000	1000	0−−	30.57	11.71	18.86	305,700.00
73	10000	5000	0−0	30.56	11.70	18.86	305,600.00
73	10000	10000	0−+	30.56	11.70	18.86	305,600.00
73	50000	1000	00−	25.09	6.23	18.86	1,254,500.00
73	50000	5000	0	25.08	6.22	18.86	1,254,000.00
73	50000	10000	00+	25.08	6.22	18.86	1,254,000.00
73	100000	1000	0+−	24.41	5.55	18.86	2,441,000.00
73	100000	5000	0+0	24.40	5.54	18.86	2,440,000.00
73	100000	10000	0++	24.40	5.53	18.86	2,440,000.00
74	10000	1000	+−−	30.63	11.76	18.86	306,300.00
74	10000	5000	+−0	30.62	11.75	18.86	306,200.00
74	10000	10000	+−+	30.61	11.74	18.86	306,100.00
74	50000	1000	+0−	25.14	6.28	18.86	1,257,000.00
74	50000	5000	0	25.13	6.27	18.86	1,256,500.00
74	50000	10000	+0+	25.13	6.27	18.86	1,256,500.00
74	100000	1000	++−	24.46	5.59	18.86	2,446,000.00
74	100000	5000	0	24.45	5.58	18.86	2,445,000.00
74	100000	10000	+++	24.45	5.58	18.86	2,445,000.00
75	10000	1000	+−−	30.67	11.80	18.86	306,700.00
75	10000	5000	+−0	30.66	11.79	18.86	306,600.00
75	10000	10000	+−+	30.66	11.79	18.86	306,600.00
75	50000	1000	+0−	25.19	6.33	18.86	1,259,500.00
75	50000	5000	0	25.18	6.32	18.86	1,259,000.00
75	50000	10000	+0+	25.18	6.32	18.86	1,259,000.00
75	100000	1000	++−	24.51	5.64	18.86	2,451,000.00
75	100000	5000	0	24.50	5.63	18.86	2,450,000.00
75	100000	10000	+++	24.49	5.63	18.86	2,449,000.00

Integrating the production cost results in the cost summary, the production cost parameters (number of stitches, annual volume and batch size) were altered to see its effect on the welding cost of the door and the Annual Fully Burdened Cost of the RLW process. The production cost results are shown in Scenario 5.

4.7.6 RLW Design and Production Cost Scenarios

Scenarios are carried out to validate the algorithm used for calculating the design and production cost of the PPR Cost Estimation Framework. Various parametric changes were made see its effect on design and production cost values. In this section, Scenarios 1 to 4 are designed to test the cost model created for capturing design costs. The values used are data collected from the RLW Navigator project based on design iterations during the product design stages. Scenario 5 is designed to test the robustness of the RLW process developed in aPriori workbench. Batch size and annual volumes were reasonably selected using increamental values based on assumption but the number of stitches were chosen based on industrial requirements on the project.

Scenario 1

Table 4.11 describes a scenario where cost value changes were made at the product design stages of an NPI process. Originally, the design time for Halo, Door Inner Panel and Door Fixture were 180, 270 and 300 respectively which show indicated Component Design Cost values for Halo, Door Inner Panel and Door Fixture as £3750, £5625 and £6250 respectively. However, as design and inspection times changed, the cost model reflects the changes in time for the above components design into cost as highlighted in Table 4.11.

Table 4. 11: (Scenario 1) Product Component Design and Inspection Time Changes

PRODUCT DESIGN COST										
Component Design Cost							Inspection Cost			
Component Name	Design Time	Rework Time	Designer's Rate	Rework Cost	Design Cost		Component	Inspection Time	Inspector's Rate	Cost
Halo	216	5	20.83	104.17	4500.00		Halo	20	24.48	489.58
Door Inner Panel	400	6	20.83	125.00	8333.33		Door Inner Panel	58	24.48	1419.79
Door Fixture	500	12	20.83	250.00	10416.67		Door Fixture	105	24.48	2570.31
Latch Re-inforcement	40	4	20.83	83.33	833.33		Latch Re-inforcement	5	24.48	122.40
Hinge Re-inforcement	70	3	20.83	62.50	1458.33		Hinge Re-inforcement	10	24.48	244.79
Hinge Plate	30	3	20.83	62.50	625.00		Hinge Plate	10	24.48	244.79
Belt Reinforcement	30	4	20.83	83.33	625.00		Belt Reinforcement	10	24.48	244.79
Window Channel	60	4	20.83	83.33	1250.00		Window Channel	5	24.48	122.40
			20.83	0.00	0.00				24.48	0.00

In this scenario, the design time and inspection times were changed as indicated in red to observe the overall effect on the total design cost within the cost summary. All other cost values remained unchanged. The cost results obtained from the changes to the some of the input parameters above is shown in Figure 4.41.

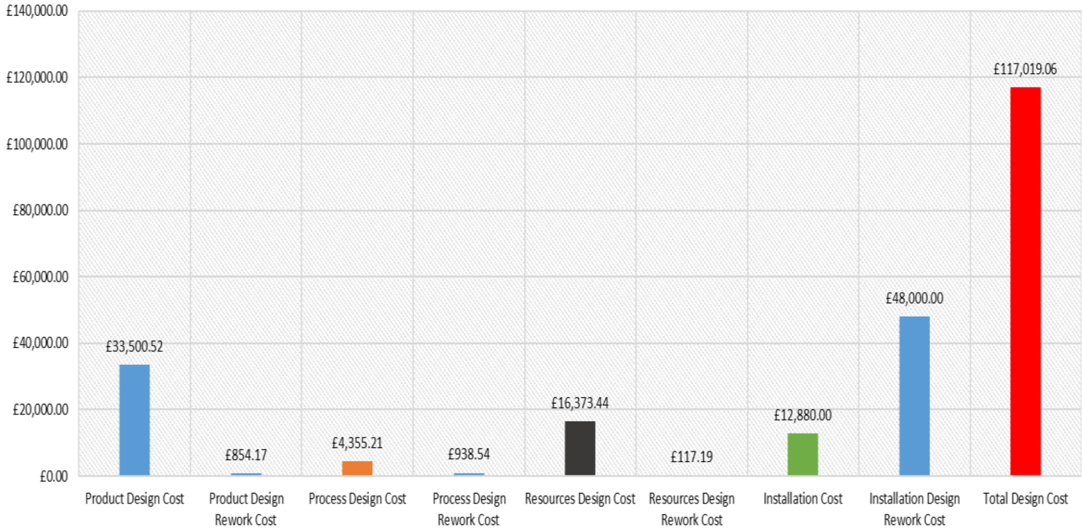


Figure 4. 40: Scenario 1 for Product Design Cost Effect on Parameter Changes

Scenario 2

Scenario 2 maintains the results obtained from Scenario 1 and makes changes to Designers' and Inspectors annual salary (£37,000 and £50,000 respectively). This changes the designers' rate to £19.27/hr and inspectors' rate to £26.4/hr. Also, extra redesigns were done for ST130 and ST140. Table 4.12 shows the input changes made in this scenario in red text.

Table 4. 12: (Scenario 2) with workstations ST130 and ST140 rework times increased

PROCESS DESIGN COST											
Process Design Cost							Inspection Cost				
WorkStation Name	Design Time	Rework Time	Designer's Rate	Rework Cost	Design Cost		WorkStation	Inspection Time	Inspector's Rate	Cost	
ST100	10	10	27.60	276.04	276.04		ST100	2	31.25	62.50	
ST110	40	4	27.60	110.42	1104.17		ST110	4	31.25	125.00	
ST120	8	4	27.60	110.42	220.83		ST120	2	31.25	62.50	
ST130	48	20	27.60	552.08	1325.00		ST130	7	31.25	218.75	
ST140	15	60	27.60	1656.25	414.06		ST140	10	31.25	312.50	
ST150	4	3	27.60	82.81	110.42		ST150	1	31.25	31.25	
ST160	4	3	27.60	82.81	110.42		ST160	1	31.25	31.25	
ST170	5	3	27.60	82.81	138.02		ST170	2	31.25	62.50	
			27.60	0.00	0.00				31.25	0.00	

The results of the input changes are shown in Figure 4.2.

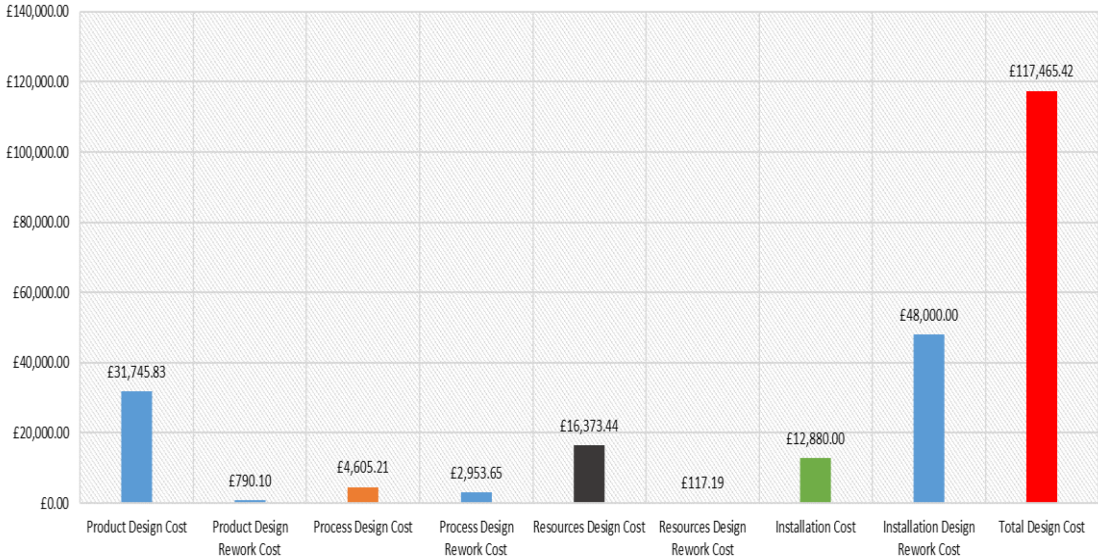


Figure 4. 41: Scenario 2 for Product Design Cost Effect on Parameter Changes

Scenario 3

Scenario 3 maintains the results from 2 but made changes to designers and inspectors annual salary, hence designers' rate and inspectors' rates are changed to £19.27/hr and £26.04/hr respectively as shown in Table 4.13.

Table 4. 13: (Scenario 3) Process Workstations Design and Inspection Time Changes

PROCESS DESIGN COST										
Process Design Cost							Inspection Cost			
WorkStation Name	Design Time	Rework Time	Designer's Rate	Rework Cost	Design Cost		WorkStation	Inspection Time	Inspector's Rate	Cost
ST100	10	10	27.60	276.04	276.04		ST100	2	31.25	62.50
ST110	40	4	27.60	110.42	1104.17		ST110	4	31.25	125.00
ST120	8	4	27.60	110.42	220.83		ST120	2	31.25	62.50
ST130	87	4	27.60	110.42	2401.56		ST130	16	31.25	500.00
ST140	120	3	27.60	82.81	3312.50		ST140	40	31.25	1250.00
ST150	10	3	27.60	82.81	276.04		ST150	4	31.25	125.00
ST160	8	3	27.60	82.81	220.83		ST160	3	31.25	93.75
ST170	8	3	27.60	82.81	220.83		ST170	3	31.25	93.75
			27.60	0.00	0.00				31.25	0.00

The results of the changes are shown in Figure 4.43.

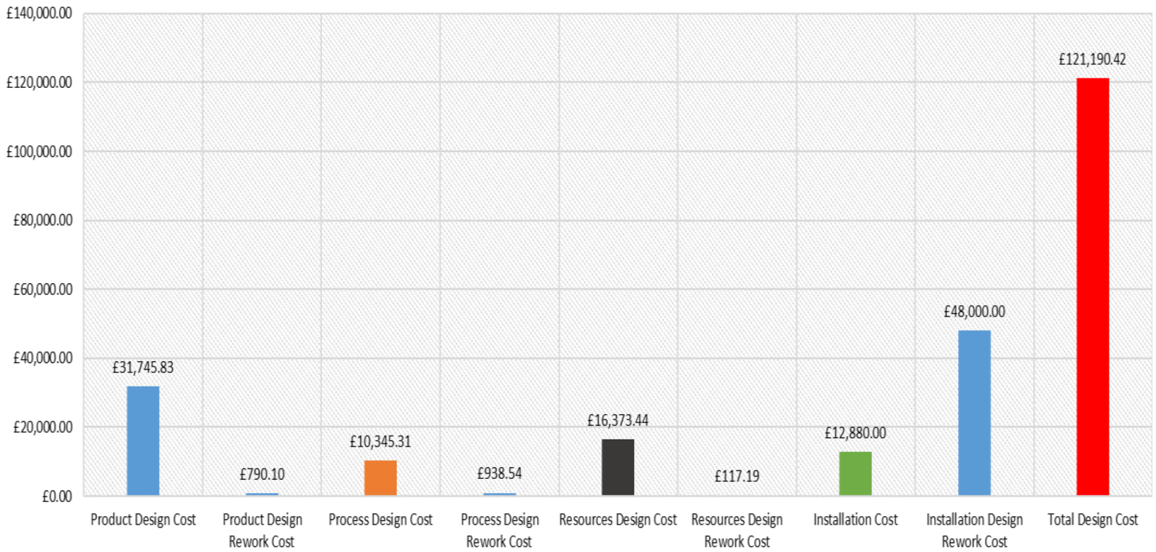


Figure 4. 42: Scenario 3 for Product Design Cost Effect on Parameter Changes

Scenario 4

In Scenario 4, results obtained in scenario 3 were maintained but changes were made to installation cost drivers. Table 4.14 shows the changes made with the installation times.

Table 4. 14: Scenario 4 Installation rate and Time Changes

INSTALLATION COST					
Engineer	Rate	Time	Rework Time	Rework Cost	Design Cost
Mechanical	30	260	130	33800	7800
Electrical	30	200	60	12000	6000
IT	35	110	110	12100	3850
				0	0
				0	0

The results obtained in scenario 4 is shown in Figure 4.44.

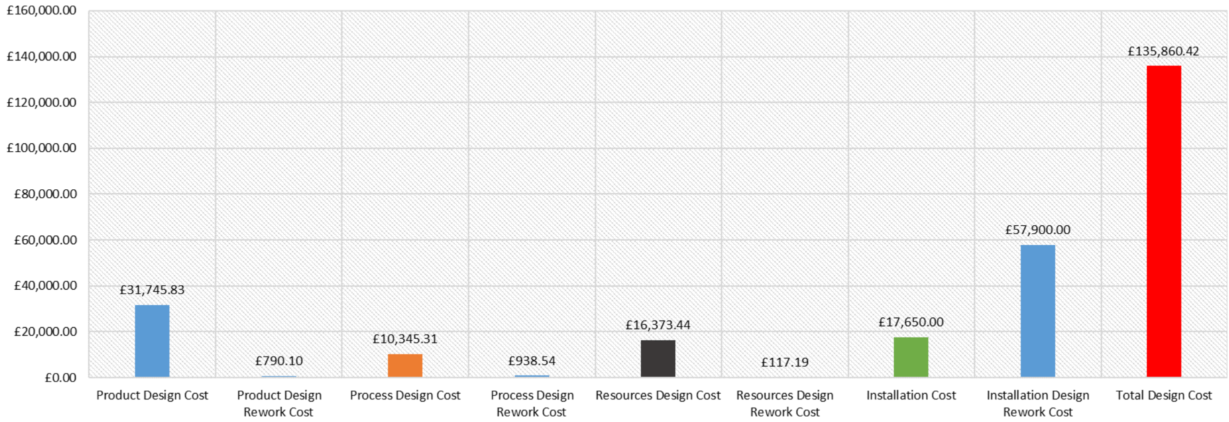


Figure 4. 43: Scenario 4 for Product Design Cost Effect on Parameter Changes

Scenario 5

Although the production cost of the welding process can be calculated manually, many parameters that relates to fixed and variable costs may have to be considered such as floor space, energy consumption, labour cost, material cost, robot cost, etc. there may be human errors of computation and inconsistencies which may reflect on the final cost value. There using a computerised system may reduce or eliminate such issues, hence producing a much better result. This scenario therefore uses the developed cost model in addition to the RLW process implemented in the aPriori cost engine to generate cost results which is faster and less tedious compared with manual cost calculation.

This scenario considers the production cost where cost parameters such as annual volume, batch size and the number of weld stitches are changed to see its effect on the RLW welding process cost and the Annual Fully Burdened Cost of the RLW process.

A. With the lowest values of annual volume of 10000, batch size of 1000 and 70 weld stitches, the production cost values are shown in Figure 4.45

Production Cost Table

Cost Type	Value
Annual Volume	10,000.00
Batch Size	1,000.00
Assy Process Fully Burdened	11.56
Sub Comp. Fully Burdened	18.86
Full Burdened Cost	30.42

Integrated Cost Table

Product Design Cost	£31,537.50	55.60%
Process Design Cost	£7,534.90	13.28%
Installation Cost	£17,650.00	31.12%
Total Design Cost	**£56,722.40**	**100.00%**
Annual Fully Burdened Cost	**£304,200.00**	

Number of Stitches	Annual Volume	Batch Size	RLW Welding Cost
70	10000	1000	30.42

Figure 4. 44: Scenario 5 (A) RLW Production Cost Effect on Parameter Changes

B. With the highest values of annual volume of 100000, batch size of 10000 and 70 weld stitches, the production cost values are shown in Figure 4.46

Production Cost Table

Cost Type	Value
Annual Volume	100,000.00
Batch Size	10,000.00
Assy Process Fully Burdened	5.39
Sub Comp. Fully Burdened	18.86
Full Burdened Cost	24.25

Integrated Cost Table

Product Design Cost	£31,537.50	55.60%
Process Design Cost	£7,534.90	13.28%
Installation Cost	£17,650.00	31.12%
Total Design Cost	**£56,722.40**	**100.00%**
Annual Fully Burdened Cost	**£2,425,000.00**	

Number of Stitches	Annual Volume	Batch Size	RLW Welding Cost
70	100000	10000	24.25

Figure 4. 45: Scenario 5 (B) RLW Production Cost Effect on Parameter Changes

C. With the lowest values of annual volume of 10000 and batch size of 1000 for 75 weld stitches, the production cost values are shown in Figure 4.47

Production Cost Table

Cost Type	Value
Annual Volume	10,000.00
Batch Size	1,000.00
Assy Process Fully Burdened	11.8
Sub Comp. Fully Burdened	18.86
Full Burdened Cost	30.66

Integrated Cost Table

Product Design Cost	£31,537.50	55.60%
Process Design Cost	£7,534.90	13.28%
Installation Cost	£17,650.00	31.12%
Total Design Cost	**£56,722.40**	**100.00%**
Annual Fully Burdened Cost	**£306,600.00**	

Number of Stitches	Annual Volume	Batch Size	RLW Welding Cost
75	10000	1000	30.67

Figure 4. 46: Scenario 5 (C) RLW Production Cost Effect on Parameter Changes

D. With the highest values of the annual volume of 100000 and batch size of 10000 for 75 weld stitches, the production cost values are shown in Figure 4.48.

Production Cost Table

Cost Type	Value
Annual Volume	100,000.00
Batch Size	10,000.00
Assy Process Fully Burdened	5.63
Sub Comp. Fully Burdened	18.86
Full Burdened Cost	24.49

Integrated Cost Table

Product Design Cost	£31,537.50	55.60%
Process Design Cost	£7,534.90	13.28%
Installation Cost	£17,650.00	31.12%
Total Design Cost	**£56,722.40**	**100.00%**
Annual Fully Burdened Cost	**£2,449,000.00**	

Number of Stitches	Annual Volume	Batch Size	RLW Welding Cost
75	100000	10000	24.49

Figure 4. 47: Scenario 5 (D) RLW Production Cost Effect on Parameter Changes

4.8 Integrated Cost Estimation for RLW

The RLW welding process cost data obtained in section 4.8.4 are integrated with the PPR Cost Estimator to give an integrated cost data for design and manufacturing. This makes cost values available and accessible to both design and manufacturing.

Therefore, the integrated cost calculator combining all five scenarios as shown in Figure 4.49 consists of Product design, Process design, Resource design, Installation and Production cost. This gives a summary of the cost using the RLW technology, where engineering changes made that relates with time instantly affects the total cost.

Figure 4. 48: RLW Integrated Cost Calculator

Various scenarios were conducted to ensure that cost parameters changes made at any point has direct causality on the overall cost of design and or manufacturing which are automatically displayed graphically.

4.9 A Comparative Analysis of Resistant Spot Welding and Remote Laser Welding process

To be able to assess and validate the output of the methodology, RSW and RLW processes are compared with each other. In practice, cost and process information is readily available for RSW as it is the current technique of fabricating care doors in the automotive industry today. However, the methodology developed shows how cost information was extracted for the RLW technology by creating models that capture the process setup required for using the technology. The comparison between the use of the two technologies focuses on the welding process cycle times and the welding process costs.

4.9.1 Process Cycle Times: RSW v RLW

According to industry experts and literature, (RSW) spot weld nugget ranges from 3mm to 8mm depending on the thickness of the sheet metal to be welded. In this research, 3mm weld nugget was used based on the material thickness and the composition of the material to be fabricated. The selected nugget size was determined by industrial experts who were part of the RLW Navigator project. Using aPriori to make 140 spot welds on the door with annual production volume of 100,000 and allowing aPriori to predict the batch size, it takes 414.40 seconds which is about 6.91 minutes to complete the welding process on each door. This can further be translated into the system requiring 2.9 seconds per spot weld as shown in Figure 5.4. Industrial experts confirmed that each spot weld takes 3.5 seconds working with 40 weld spots.

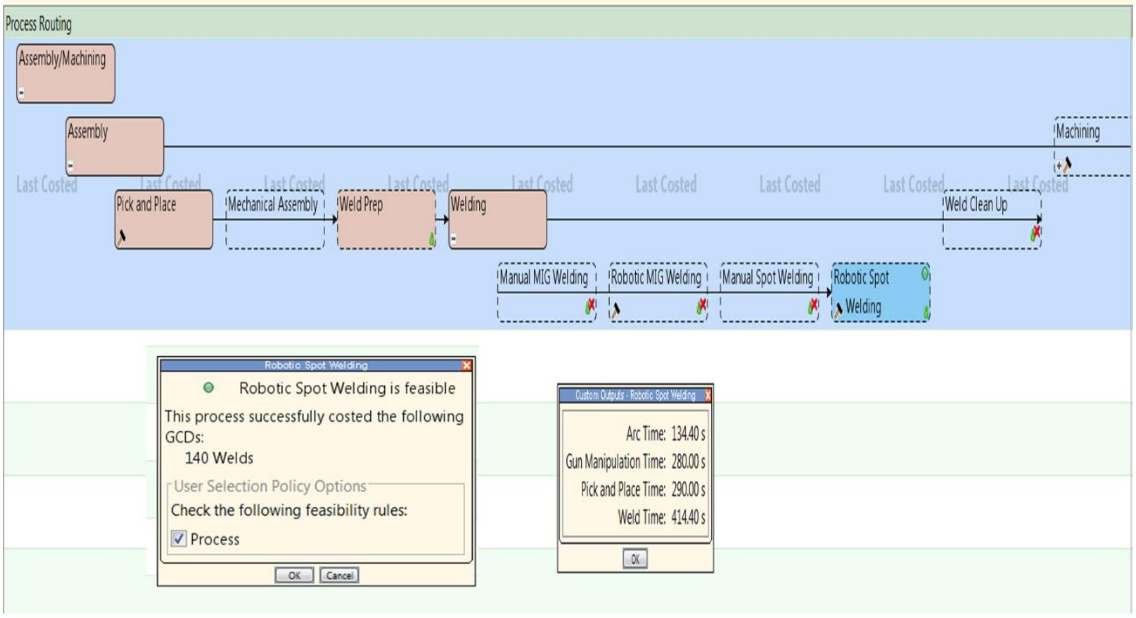


Figure 5. 1: RSW process time for 140 spot welds with an annual volume of 100,000 using aPriori default batch size.

However, there is a vast difference between cycle times obtained from aPriori using the RLW process. Considering the maximum number of welds which is 75 stitches on the door, with an annual production volume of 100000 and having a batch size of 10000, the cycle time required is 36.37 seconds. This indicates that the system requires 0.6 seconds to make a weld stitch on the door as shown in Figure 5.2.

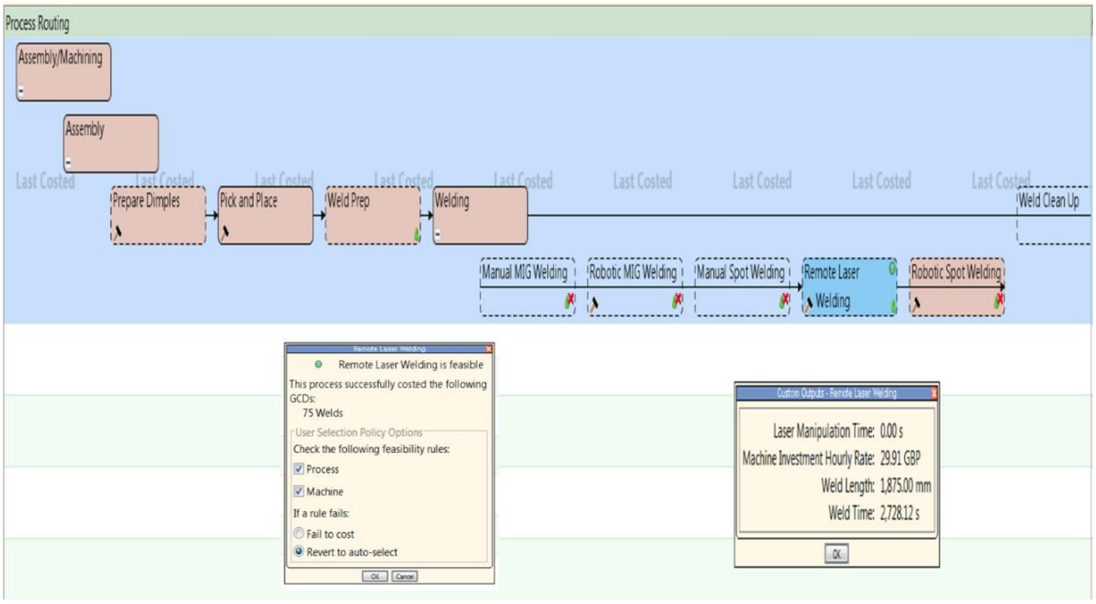


Figure 5. 2: RLW process time for making 75 stitches with an annual volume of 100,000 using a batch size of 1,000.

Comparing the two cycle times, it is obvious that RLW technology has a better process cycle time compared with the current RSW technology. This is due to the fact that RLW does not require physical contact with the door but rather access the welding area remotely about 1m away which make it easier and faster to make the weld stitches on the door.

4.9.2 Welding Process Costs: RSW v RLW

The welding process cost for RSW using the same parameter above in aPriori shows that the cost making 140 weld spots using resistant spot welding technology is about £6.55 per a complete welding process. This indicates that the cost per spot weld is approximately £0.05. This is the cost generated based on aPriori's inbuilt cost information containing specific resource and process data for the resistant welding process. The process summary of RSW in aPriori is shown in Figure 5.3, showing the assembly process and the cost of the process.

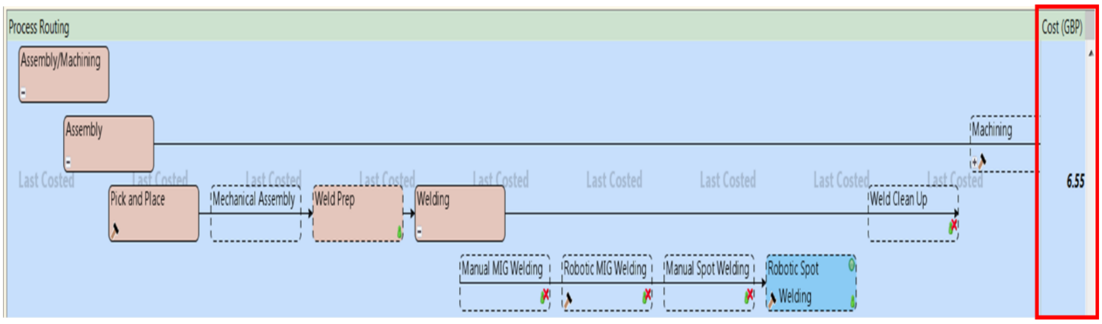


Figure 5. 3: RSW process cost for 140 spot welds with an annual volume of 100000 using aPriori batch size.

However, according to the industry experts in the automotive industry, the cost per spot weld ranges between £0.010 and £0.16. Industrial costs were generated based on the cost of Quality, Maintenance, Spare Parts, Process Consumables, Services Consumables, Production Area, Implementation Costs, Process Quality Control, Hardware, Incoming Services and Engineering costs.

RLW cost was generated from the developed process in aPriori for a production annual volume of 100,000 with a 1,000 batch size realized a cost of £5.63 to complete 75 weld stitches. This further indicates that each weld stitch costs £0.075 as shown in Figure 5.4. It is significant that the cost value obtained per stitch from RLW is lower compared to the existing spot welding technology due to the fact that fewer resources (robots) were required for the welding process.

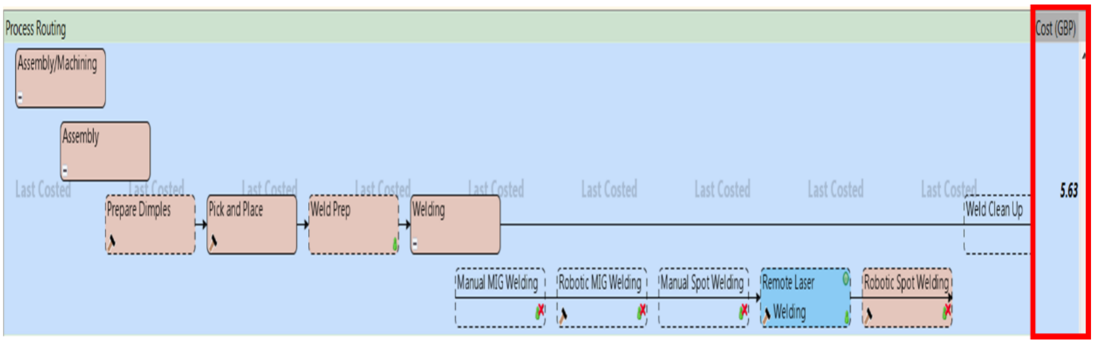


Figure 5. 4: RLW process cost for making 75 weld stitches with an annual volume of 100,000 using a batch size of 1,000.

4.10 Summary

PPR Cost Estimation Framework has been used in the RLW process to demonstrate its application in using real data where the step by step approach explained in Chapter 3 were followed. The case application considered the RLW product (3D model of a car door) as an input to the process received from product design. The product was modelled according to the requirements of the methodology, where; a product tree was created to show the how product components are to be introduced into the process; a computer representation was generated to support further application of the product tree and then cost calculation algorithm that calculated the total cost for designing the complete door. Similarly, the process model was developed for the RLW process which contained a graphical process and workstations illustration; a computer representation of the processing logic and the cost calculation algorithm for estimating the cost of designing the manufacturing process. Furthermore, a resource illustration, representation, illustration, database and cost estimation calculation were developed for the resource model. The PPR cost calculator was then used for integrating all the cost values for the product, process and resource models in a common database to make cost values available to all designers in a form of a cost dashboard.

The PPR models developed were integrated with a cost modeller (aPriori) as an interrogation tool for decision support, showing how the process capabilities of aPriori were extended to include the RLW process for calculating welding process cost on a car door. The manufacturing cost obtained were integrated with the integrated cost calculator to predict the cost of design, installation and manufacturing using RLW technology as shown in Figure 4.49.

CHAPTER 5

DISCUSSION

In this chapter, a review of the proposed PPR Cost Estimation Framework will be conducted. A discussion of the results obtained from the use of the methodology on the RLW Navigator project will also be carried out in a way that shows the generic application of the methodology. Also, a comparative analysis of the proposed methodology's results on RLW with resistant spot welding (RSW) technology will also be done in this chapter. Then, the use of PPR Cost Estimation on other new processes will be discussed as well as how to apply the framework in general. Finally, its benefits, requirements and limitations of the framework will be done.

5.1 A Review of The Proposed PPR Cost Estimation Framework

As described earlier in Chapter 3, the proposed PPR Cost Estimation Framework consists of three parts;

- A Product-Process-Resource Modelling Technique for Capturing Engineering Knowledge and Cost Values
- A technique for extending cost modeller capabilities to include a new process for cost assessment and
- A Technique for Integrating P-P-R-Production Cost Values to Support Engineering Decisions

A Product-Process-Resource modelling technique for capturing engineering knowledge and cost values – as seen in literature (Chapter 2) current research demonstrates various techniques for modelling enterprise for representing and describing elements and sub-elements. These techniques, however, support in minimizing enterprise design complexity, increase coherence, align business and IT, analyze operations and optimize and re-engineer both enterprise structure and behaviour (Fayoumi 2016). In spite of the wider benefits addressed by some of the systems modelling techniques, it was observed that all the techniques rely on already existing and detailed structured data for modelling. It is shown in this research however that, engineering knowledge can be captured through an integrated product-process-resource modelling technique that demonstrates and represents a new system with fewer data. This was achieved by creating lower level sub-processes (workstation) that are integrated with resources that are capable of realizing the product. It was also observed that although a lot of research is done in the domain of enterprise modelling and cost estimation, very little attention has been given to the integration of the two. To fill this gap in research, a cost model was developed that integrates existing cost accounting algorithms into a unique cost estimation calculator for generating cost values for a product, process and resource design cost during early design stages. Cost information is extracted from the detailed integrated product-process-resource models.

A technique for extending cost modeller capabilities to include a new process for cost assessment – although existing research and commercial software and proprietary tools which support manufacturing cost estimations such as FIPER (Koonce, Judd et al. 2003); TIMCES (Wong, Imam et al. 1992) and neural network-based approaches are described in (Shtub and Zimerman 1993), no research has shown how to model process and resource to satisfy the input required to extend the capabilities of a cost estimation software tool. As a result, most tools are not capable of estimating the cost of a product that requires a new manufacturing technology. To overcome this, process and resources input requirements of a commercial cost estimation tool were identified in a systematic manner and processes and resources were modelled to satisfy those requirements. Also, XML files of the models created were generated and exported into the tool to maintain consistency of the models in a computer representation. Generating a computer representation simplifies systems design and modelling process by avoiding redesign of the same system into different IT modelling tools such as DES and DS tools. In this research, only one commercial software tool was used to demonstrate this technique. Further work may be done to replicate the proposed PPR methodology with multiple commercial off the shelf tools. for. The author is of the view that more complex scenarios with other commercial software will be more interesting.

A Technique for Integrating P-P-R-Production Cost Values to Support Engineering Decisions - As identified in literature, most engineering research in cost estimation for many years use cost accounting tools and techniques. Most often, researchers either focus on developing techniques for product design cost estimation or predicting and optimizing the cost of a manufactured. Most researchers use complex mathematical algorithms which are not so simple to follow during implementation. The author is of the view that currently, no research has been identified to have a model that integrates product, process and resource design cost and production process cost values to display design and production cost summary. This proposed integrated technique is able to capture cost value changes made during design on a product component and also shows its effect on the total design and production process cost. The technique is modelled in a modular way that products, processes, resources, and production values may be added or removed easily. A useful benefit of this cost model is its ability to be used in IF/THEN analysis to support engineering decision making on cost. Although cost accountant attempts to estimate manufacturing process cost, they lack a detailed understanding of complex manufacturing processes with limited cost data association. Hence, traditional accounting systems according to Maskell (1991) has the following limitations: they lack relevance; costs are distorted; they are inflexible; they offer practical hindrances to dynamic manufacturing systems; they are subject to the needs of financial accounting.

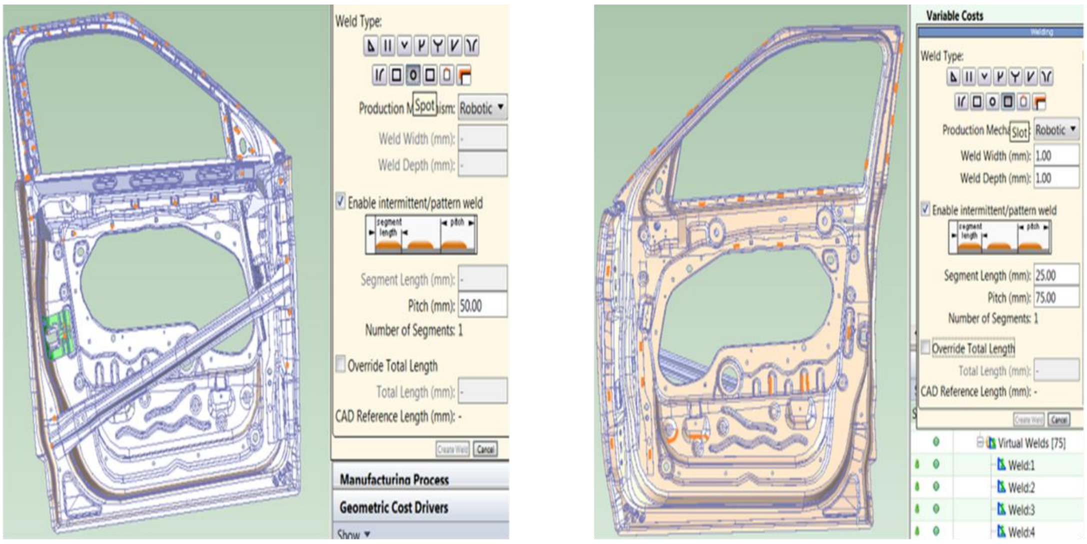
As shown in Chapter 2 of this research, currently, systems modelling techniques and cost estimation techniques are not integrated. This means that, system modellers may not consider cost as a performance indicator and likewise, cost estimators may not fully understand the dynamics of a system model. This research therefore proposes a framework to bridge the gap between the two domains. The proposed framework in this research has shown how product, process and resource data are integrated using modelling techniques. The integration in the framework was achieved by modelling the process flow by breaking it down into workstations. The workstations contains details resource activities, that shows which resources are consumed, when they are required and how long they are utilized as shown in Chapter 3. The developed models were then used to extend the capabilities of a commercial of cost modeller to include a new process. This was achieved by modelling the process logic and resources flow to meet the input requirements of the cost modeller. A script of the process and the resource models were generated and used as inputs to the cost modeller as demonstrated in Chapter 4.The fundamental mechanism of the proposed framework that is new is the developement of detailed process and resource models and generating scripts of the models that matches the input requirements of a cost modeller.

Another capability of the framework is its ability to capture the cost of product, process and resources (PPR) design at early stages. The framework achieves this by using design cost models for estimating the design costs. The design cost models were developed using standard cost accounting equations using design time, designer's rate and rework cost as inputs to the cost function. These design cost models are integrated to generate the total cost of PPR design such that, changes to any input values automatically reflects in the total cost of design.

The proposed framework being verified and validated using industrial data, the author is of the view that, the research question "How can product, process and resource be integrated in a way that ensures that engineering knowledge, product and process design costs are captured during early design stages when introducing a new technology?" is answered using the proposed PPR Cost Estimation Framework. The complete description and illustration of the framework are shown in Chapters 3 and 4.

5.2 Welding Technologies: RSW v RLW

The welding technologies used for making welds for RSW and RLW are different in the sense that with RSW, it is required that thicker parts are placed on top of the thinner part, whereas RLW requires that the thinner part goes on top of the thicker part. An illustration of this is shown in Figure 5.5, where (a) shows that spot welds are done from other components onto the door inner, as the door inner is the thinnest amongst all components. Diagram (b) also shows the use of RLW process created in aPriori, where weld stitched were created from the door inner onto other subcomponents due to material thickness.



(a) Sample of resistant spot welds made on car door (b) Sample of remote laser weld stitches made on car door

Figure 5. 5: A comparison of RSW and RLW welding techniques using aPriori

5.3 Results of Remote Laser Welding using PPR Cost Estimation Methodology

This purpose of this section is to show how unstructured data has been modelled to generate cost values for cost related decision. Also, to validate that the algorithm used for calculating the various design estimates are correct and consistent by comparing results with manual calculations using traditional cost accounting techniques. Finally, to illustrate that parametric changes made to product, process and resource design stage as well as production stage reflects instantly on the total cost estimate. To do this, the following results will be discussed:

Product Design Cost - most cost estimation tools such as aPriori, Costimator, SEER and others do a good job of taking a 3D model of a product, comparing it with a spreadsheet of data that are already stored in the tool's database and then generate a manufacturing cost. These tools, however, fail to determine the cost of designing the product to be manufactured. Also, other product costing techniques such as activity based costing (ABC), parametric costing, feature based costing and other reported techniques reported in literature do not consider the design cost of components. Hence, the author is of the view that the cost of product design is unaccounted for in major cost estimation tools. The proposed methodology, therefore, shows how the product design costs are generated through modelling of cost related variables. This is shown in the results obtained for the design of the RLW door in section 4.7.1. This was achieved by creating a cost model for the RLW product with its components that automatically sums up the cost of designing each individual component of the door. For each component, the cost of engineering design was calculated by adding the components design cost and components inspection costs as shown in Table 4.6. The costing algorithm used for calculating the product design cost is shown in section 3.2.1.1 in equations (3.1) to (3.7). In the model cost model created for the methodology, rework cost is introduced to capture the cost of engineering changes done on each individual component. However, in most cost models, rework is usually not considered during cost estimation process but in reality, rework is usually part of the design process. An estimated rework cost is also calculated by allocating potential rework time for each design and then calculating the rework cost. Cost of inspecting the designs was calculated by multiplying the inspectors' rate by the expected time to inspect the design is added to the cost of the engineering design. In Table 4.6, the design time and designer's rate can be changed to reflect the cost of a component. Table 4.7 also shows the summary of the product design cost which shows the design cost and inspection cost for the individual components of the product.

Process Design Cost - The results in Table 4.8 shows how unstructured process data is modelled to extract process cost information. In most cases, process design costs are estimated by experts based on their experience on similar processes. Very limited literature is available on how expert's knowledge is captured during early process design cost estimation. Current process estimation tools and techniques cost the process after the process design stage. The use of the proposed PPR Cost Estimation methodology has shown how design cost of a process is estimated using SysML modelling techniques to illustrate the process of the novel RLW process to help gain a better understanding of the process and also, to extract cost values for estimation. The cost model developed (Figure 4.7) for calculating the process design cost uses the cost algorithms displayed in section 3.2.1.3 in equation (3.8) to (3.13). Using the developed cost model, each workstation's expected design time is multiplied by the process designer's rate. The inspection cost for each workstation's design is added to the workstation design cost. So, for example, the design cost for ST100 is £338.54, which is the sum of design cost of £276.04 and inspection cost of £62.50 as illustrated in Table 4.9. Hence, the sum of all workstations design cost gives the total cost for the designing the process.

Resource Design Cost – the resource design cost was calculated in the case application using cost algorithms (3.14) to (3.18) displayed in section 3.2.1.3. A cost model was developed for the calculation which considers the design cost of each resource required by the process for product realisation. For the Navigator project, the resources involved were mainly various types of robots and fixtures. The design mainly considered the flow sequence of the resources, trajectory planning for the various robots to ensure synchronisation for all the workstations within the process. The resource fixtures design also generated cost due to the fact that it is fully automated and required to operate in synchrony with all the robots to ensure that the overall workstation and process cycle times are met. As illustrated in Table 4.8, design times for the resources were generated through resource design simulation by experts on the Navigator project with some assumptions made for the rework times.

Installation Cost - The installation cost for the RLW Navigator project required Mechanical, Electrical and IT functions. The installation cost used equations (3.22) to (3.28) for developing the cost model as shown in (Figure 4.9). For the purpose of this research, the installation cost only considered mechanical, electrical and IT costs. However, other cost factors may be added to the cost model to generate cost values. Summing up the costs for all the functions for the Navigator project gives the total design cost estimate for installation of £12,880 as shown in Figure 4.9.

Remote Laser Welding Process Cost - The cost of the welding process was based on several parameters such as the annual production volume, batch size, number of stitches, stitch length, stitch width, material type and production life. The use of existing cost modellers does not give users the opportunity to modify existing cost logic for customising built in this research, a demonstration of how the logic can be modified is demonstrated. The process cost equations used were generated as shown in equations (3.29) to (3.39) and coded into the cost modeller using cost scripting language (CSL). These parameters were selected in the cost modellers using the RLW process created to generate the weld stitches cost as shown in section 4.8.4. The design of experiment shows various parametric changes made at the production stage and its causality on the RLW process cost for welding one door and the annual production cost, which is also called fully burdened cost as shown in Table 4.11.

Integrated RLW Design and Production Cost Scenarios – as mentioned earlier, there are many estimation tools for production cost purposes. There are also costing and accounting techniques for estimating unit cost of products which are capable of forecasting production volumes. In today's research and industrial practices, very little literature is available which is interactive for calculating both design and production cost that reflects the effect of engineering changes instantly. PPR Cost Estimation methodology addresses this by introducing an integrated product, process and resource design cost together with installation and production cost. This allows cost assessment to be done on various cost models which are integrated to give a total cost value. In this research, five scenarios were carried out to ensure that the cost algorithms used for the estimation work, as well as the integration of the various design cost models effectively. The first four scenarios considered the product design, process design and installation cost estimation, where parameters are changed to see its reflection on the cost of each stage. However, scenario 5 considered the production cost, where cost parameters were changed to see its effect on the annual fully burdened cost of the production process. In Figures 4.45 – Figure 4.48, the results obtained showed the various parametric changed made to the production process and its effect on cost.

Comparing the cost calculations done for design using the cost equations with the estimates derived from the integrated cost calculator, it is obvious that the integrated cost calculator is capable of responding much easy to changes than using manually calculated techniques. The benefits of this integrated cost calculator may, therefore, be much realised in situations where there are many engineering designs of components as well as tracking engineering changes that impact cost.

5.4 How PPR Cost Estimation Framework Has Been Applied in This Research

Currently, engineers can predict the cost of a product CAD using available commercial cost estimation software. These tools are only capable of generating cost values based on process logic and resource data that are already embedded in the tool. The logic behind these tools is, a 3D product CAD model has to be imported into its environment. Then, geometric features on the CAD model are identified and processes and resources that are capable of realising those features are selected for the tool's library. The cost of manufacturing the product is generated by identifying the cycle time of the process selected and multiplying the time by the resource rate. In such situations, cost estimation engineers are limited to the algorithms used by cost estimation tools hence, they have little or no control over the output values of the tools. This may, therefore, result in inappropriate processes and resource selections which may result in inaccurately estimating the cost of a product.

Building a robust PPR Cost Estimation framework for a larger customised industrial application will require a team of Product Designers, Design Engineers, Process Designers, Software Developers and Cost Estimators.

The author is of the view that the PPR Cost Estimation Framework is useful for Product Design and Manufacturing departments of an organisation to support with engineering knowledge capturing and consistency in cost estimation activities. To use the PPR Cost Estimation Framework, a summary of the steps required to be followed is shown in Figure 5.6.

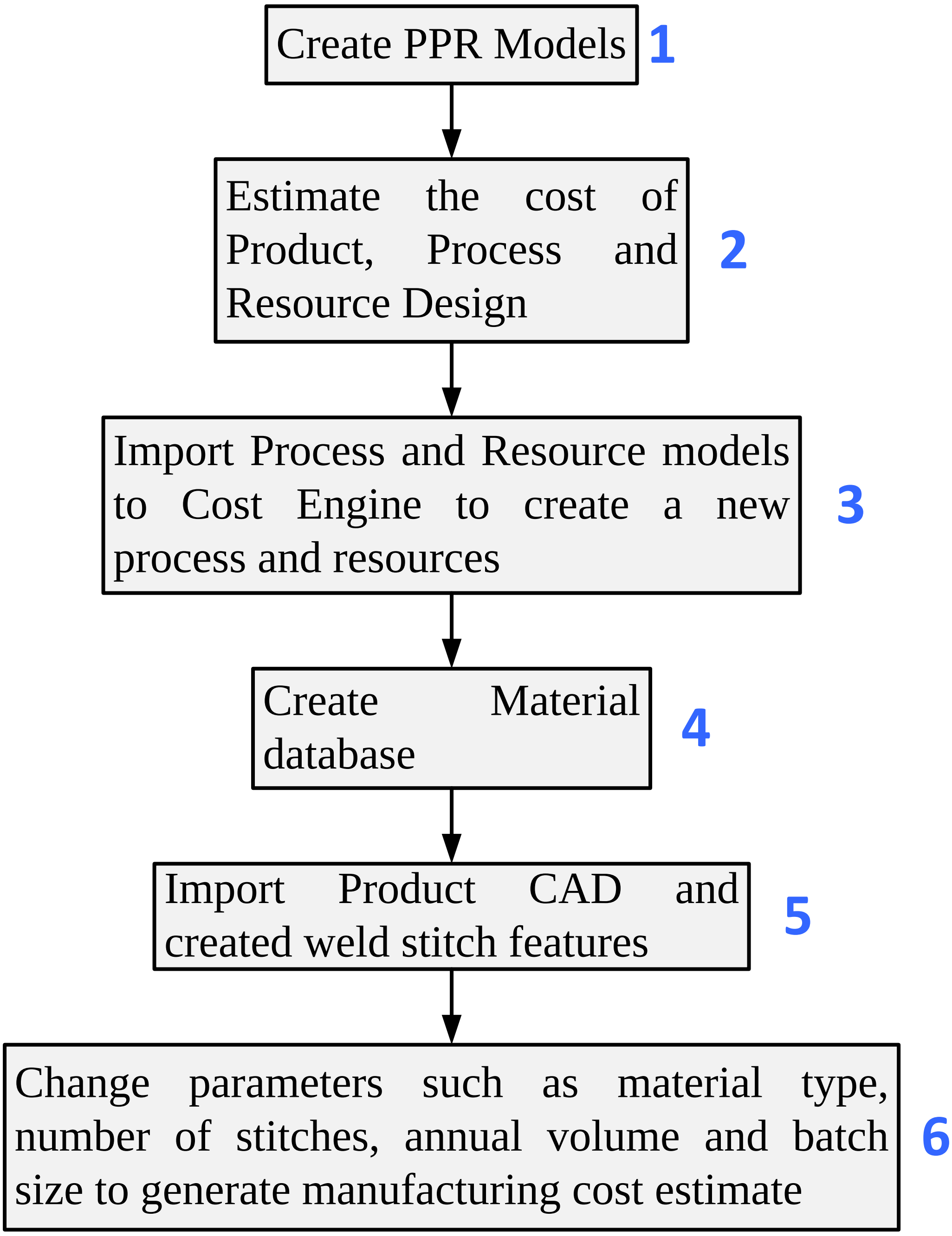


Figure 5. 6: PPR Cost Estimation Application Stages

1. **Create PPR Models:** Models must be created for Product, Process and Resources. Each model must contain the following modules: a graphical illustration, computer representation and a cost equation.
2. **Estimate Cost of Product, Process and Resource Design:** The cost of designing the product, process and resources for the system is then calculated at the next stage of the methodology. This is an estimate of the total design cost which uses the

designer's rate multiplied by the number of hours spent on the design. These values are entered in fields of the NPI design cost calculator, where a change in a value affects the total design cost of design.

3. **Extend Cost engine with Process and Resource models**: Cost modeller's engine is extended by including processes and resource that are not currently available within the cost engine. This is successfully satisfied by the models created for process and resources. Here, process logic and resource rules are created in a language compatible with the cost modeller. Furthermore, equations for calculating the new process cost is also created to link the resource rates and the process times.
4. **Create Material database:** Material data for the product was also created which contains the material name, material type, properties, material rate, etc.
5. **Import Product CAD and create weld stitches:** Product CAD model is imported into the virtual production environment where the newly created process is selected for creating weld features on the 3D CAD model. Weld stitches are defined on the 3D product CAD model and recognized as a feature on the product CAD.
6. **Make parametric cost estimates:** Virtual experiments are carried on for making weld stitches from 70 to 75 welds and varying batch sizes and annual production volumes to see its implications on cost. Cost values obtained can be used for engineering decision making.

There are three modules of the PPR Cost Estimation framework which are; a graphical representation for helping product and process designers have an overview and a better understanding of the product, process and resources within the system; a computer representation that is generated from the PPR graphical representation that is useful for computer systems analysis such as discrete event simulation (DES) and dynamic simulation analysis by avoiding redrawing of a complex system from static models. The computer representation also serves as input to cost modeller for extended manufacturing process options; cost equations, for calculating the cost associated with designing product components, process workstations design and resource design.

As a cost modelling methodology, PPR Cost estimator integrates the cost of designing product, process and resources for a given system, taking into consideration the rate of designer and the expected duration to complete the design in a PPR design cost database. An installation database is also created which contains the installation costs associated with setting up a new production system. The installation cost is a sum of mechanical engineering cost, electrical engineering cost and IT engineering cost. However, other costs associated with installation may be included where necessary. Furthermore, production cost values generated from cost modeller are extracted into a production cost database. Then, an intelligent cost algorithm is developed to integrate all the databases created to project a cost dashboard, where engineering changes made are translated into cost with its causality automatically reflecting on the cost dashboard graphically.

5.5 PPR Cost Estimation Benefits against Required Effort

Choosing to use the PPR Cost Estimator or not requires the user to consider the benefits that the methodology brings against the required efforts to realize the benefits. In this section, the benefits and the requirements of the framework are discussed.

5.5.1 Benefits

Some identified benefits of the proposed PPR Cost Estimation Framework includes:

- Enhancing design time for a new product introduction process by introducing a cost as a KPI in the form of a cost calculation dashboard that displays cost values to both design and manufacturing.
- The methodology may interface with other product lifecycle management (PLM) tools to support engineering decision making during early design stages.
- It is capable of breaking down complex systems processes into simple operations and at the same time show the integration between the process operations and resources that acts on them. This helps with the having a better understanding of a system,s process and resource interactions.
- Useful for engineers or senior management who wants to have a overview of the cost of a project that requires the design and manufacturing of multiple

components.

5.5.2 Requirements

To fully achieve the benefits of the PPR Cost Estimation framework, the following requirements must be met:

- PPR modelling and
- PPR Models implementation

PPR modelling – this requires a great effort of creating a static graphical representation, computer integrated codes and cost algorithms for product, process and resources. In addition, resource and the material database have to be created. The creation of the process and resource integration code is the backbone of the methodology because it has to compatible with the software tool for a seamless integration. Likewise, the database formats have to be tailored to the tool's input requirements.

PPR Models implementation – this requirement demands knowledge, data and the ability to expand the existing capabilities of a commercial of the shelf (COTS) software to include new process technology through integration of the created models. These include:

- A working knowledge of the software tool
- The data structure of the workbench and how to extend it
- Understanding how process and resources are integrated into operations and workstation levels
- Understanding of resource selection criteria and rules as well as how costs are generated through resource consumption
- Knowledge of the coding language required for creating and implementing process and resource logic into the software engine.
- The ability to create cost equations that integrates process, resources and materials parameters that are compatible with the tool's data structure.

It is important to mention that although some models created meet the software tools data structure requirements, hence directly implementing models into the tool's engine, other data may require manual computation which may be a time-consuming exercise. It is true that at the early design stage of NPI process, less data is available, however, the framework contains a systematic structure that allows new data to be integrated or a modification of existing data at any point for cost assessment. Nevertheless, the framework allows a minimum required data to generate cost estimates, however, the more accurate data available, better cost estimate values may be attained.

5.6 Remote Laser Welding Characteristics

Some of the noted benefits for the RLW process when compared with conventional RSW include increased flexibility in weld design as only one side access to welding area is required; elimination of direct tool to work piece contact; better weld quality obtained in terms of weight (reduction of flange sizes); high stiffness and low thermal distortion (Davies 2012; Erdős et al. 2013; Ceglarek et al. 2015). There is however a high risk of intensive laser beam (exceeding 106 W/cm2) hence, the technology requires a dedicated enclosed cell. The cell usually contains an industrial robot with end effector, laser source, cooling device, fixture device and workpiece to be welded. The robot control for the welding activity is done outside the dedicated cell. For this research, the cell is designed such that welding operation cannot be carried out when there is an operator in the cell or when the cell door is not properly shut. The enclosed and confined workspace of RLW process brings challenges for robot programming, accessibility and collisions checking. Colleagues in the FP7 RLW Navigator project (Erdős et al. 2013) have conducted research to help offline cell configuration design, robot path planning and optimisation, accessibility analysis and collision checking.

Other researchers are of the view that RLW technology generates high productivity compare with MIG welding and RSW. Grupp et al.(2003) and Duggan Manufacturing (2010) claims that due to the scanner head attached to the the end effector of the robot, the technology reduces non productive time and therefore is capable of reducing the welding process cycle time by up to 80%. Furthermore, the impelementation of the RLW technology on the Ford Mustang revealed the following efficiencies:

- The actual weld time of body-side outer assembly welds is 5X shorter, taking 22 seconds instead of 111 seconds for 79 welds;
- Single-sided access enables handling in one station cycle instead of three station cycles;
- Only four welding robots are required instead of 12;
- Only two material handling robots are required instead of four; and
- Reduced plant floor space due to the reduced number of resources and equipments used (Matthew and Christine, 2017).

5.7 Framework Limitations

Considering the research objectives, resources assigned to the research and the time limitations to complete a PhD study, the author has identified potential limitations of this research methodology in the following areas:

- The framework only considered direct operational cost elements, hence simplifying the costing process. There were other indirect cost elements that can be considered at early design stage that are key determinants of actual business process cost such as breakdown maintenance, human factors (absence, sick, holidays, etc.), environmental, health and safety, quality, etc. which were not included. Although these have direct implications on cost, the framework is simplified as much time and effort are required to capture such data to create models that capture knowledge for estimating cost.
- The methodology only made us of SysML and MS Visio for capturing engineering knowledge but there are other modelling tools and techniques such as Stella, UML, Visual Paradigm, CIMOSA, PERA, IDEFs, etc. that may be exploited. The author used the selected tools due to its availability to the research group, cost and training available. Other tools may be more useful to the methodology but each modelling tool has its own constructs which thorough understanding in order to generate meaningful models that are useful to the methodology.
- The framework can generate cost information at three levels; designing cost

information, installation and production cost information for a system. Currently, the framework automatically integrates design and installation cost databases which are semi-automatically integrated with the production database. Production values obtained from cost modeller are populated into a spreadsheet as a database. This spreadsheet is maintained and updated based on production requirements due to parametric changes into the cost calculator to reflect the total manufacturing cost of the system.

- The framework has only been applied to sheet metal welding application using experimental research environment with expert’s opinion on the validity of data used and the framework’s approach. Applying the framework in a real industrial environment data will be interesting.
- Only aPriori cost estimation software was used to demonstrate how existing tool’s capabilities can be expanded. This is because most 3D cost estimation tools are commercial and they are expensive to purchase license and training. License and training were available for aPriori to access its database to expand its capabilities.
- The detailed steps required to fully benefit from the framework are many and may seem complicated but the author is of the view that oversimplification of systems modelling and cost modelling techniques may result in models that are not a representation of a real system.

5.8 Summary

A review of the proposed PPR Cost Estimation Framework was discussed to highlight how the various components of the framework are interlinked and used. The framework was applied to the RLW Navigator project to verify and validate its robustness and its generic application. Furthermore, a comparative analysis of the proposed methodology's results on RLW with resistant spot welding (RSW) technology was carried out. Then, a summary of how the PPR Cost Estimation Framework has been used in this research was highlighted, looking at how the methodology can be applied to innovative processes. Finally, the benefits, requirements and limitations of the PPR Cost Estimation Framework were discussed.

CHAPTER 6

CONCLUSION AND FURTHER WORK

This chapter concludes the research by reflecting on the development and application of the proposed methodology. Research contributions to the body of knowledge in the domain of manufacturing are also addressed. Furthermore, recommendations and future works that are capable of enhancing the performance of the methodology are given. Finally, the author gives concluding remarks on the research.

6.1 Conclusion

The research presents a novel methodology (PPR Cost Estimation Framework) for estimating cost based on product and process features. It is based on the integration of product features with a feasible process and resource solutions. The motivation for the derivation of the methodology was based on observed gaps in the literature and the over-reliance on technical KPIs with little focus on the economic implication of technical decisions. Also, the introduction of RLW on sheet metals introduces new design challenges as conventional joining processes have heavily relied on product features suitable for Resistance Spot Welding (RSW), Self-Piercing Riveting (SPR) and adhesive technologies. Because of the economic benefits achieved through the RLW technology (50-75%-reduced processing time; 50% decreased factory-floor space and 60% reduced environmental impact when compared with its competitive technology, RSW), the industrial-wide application will require the redesign of product features to ensure remote laser welding feasibility. This is because laser properties and accessibility, material thickness and gaps between sheet metals to be welded are critical factors for ensuring weld quality. This, therefore, generates new product design challenges such as the assessment of sheet metal weldability before laser welding implementation. Whilst addressing the technical product design challenges associated with this new joining process, the economics of design decisions (feature selection, orientation, dimensions, stitch type, locations, fixturing, etc.) must be well understood. Also, product designers must be supported with RLW process design knowledge thus reducing typical errors which often exist in product design activities as a result of the lack of knowledge integration. Although there are commercial tools available that are useful for managing processes, such as the Product Lifecycle Management (PLM) tools and Knowlwdge Management (KM) tools, there is still the lack of integration between product, process and resource design funtions in most larger organisation. This methodology attempts to bridge the existing gaps in knowledge sets emanating from these disciplines. As a result, the fundamental models rest on knowledge-based models which are supported by predictive algorithms for associating base product, process, resource and cost models.

The structure and initial application of the PPR Cost Estimation Framework have been presented in chapters 3 and 4. The cost values and process models presented in this research have not been related to any industrial product due to confidentiality and sensitivity of data. Also, the proposed methodology was applied to integrate product, process, resource and cost related to the lab-based production of a car door using the RLW technology. The author is aware of the fact that there are significant differences between industrial production methods and the lab-based methods examined. Although these differences exist, the modelling logic was carefully tested and the initial results derived from the application of the methodology was confirmed by industrial partners as being a true reflection of their expectations. It, therefore, confirms that product features can dynamically be integrated with process, resource and cost data for useful engineering and economic analysis.

This research has specifically introduced new stitch modelling features into the aPriori workbench. New cost estimation equations and rules for RLW technology have been developed through the use of the CSL language. New process and resource routings dedicated for RLW applications have also been developed and reported in this paper. Furthermore, global templates for populating specific data for RLW processes (PPR Cost Estimation) logic have been developed and used in this research. It follows that through the application of the proposed methodology on the cost modelling workbench, material, labour, piece-part, capital and operational cost related to an engineering design can be assessed. Changes to these engineering designs will generate different cost metrics and can, therefore, be used for comparative analysis.

Also by applying the method reported, processes and resources suitable for given product features can be analysed, modified and stored accordingly, for future use.

6.2 Research Contributions

This research has resulted in the following research contributions to the body of knowledge.

1) A Product-Process-Resource Modelling Technique for Capturing Engineering Knowledge and Cost Values. This technique captures knowledge that represents real instances of manufacturing processes to support cost estimation during the early design stage. Current cost estimation methods and tools are only capable of estimating the cost of product, process or resources where information and technology already exist. This is because best process modelling techniques in support of cost modelling assume single flow dedicated manufacturing systems and underestimate the implication of multiple flows, resource sharing, reworks and other manufacturing dependent failures. This methodology has the advantage of being able to model and integrate product-process-resource data that helps with the understanding of manufacturing system to capture the required knowledge and cost values that are a true representation of manufacturing activities.

2) A Technique for Extending Cost Modeller Capabilities to Include A New Process For Cost Assessment. Although there are commercially available computer-assisted cost estimation systems and other cost estimation techniques, there is still the need for expert's opinion on cost values which may be biased, inconsistent or unsuitable for purpose. Most cost estimation tools have predefined processes and resources which may not be a representative of an actual production environment. In many cases, 3D estimation tool generates costs based on geometric features of a product and users are not able to modify process and resource parameters for customization. The proposed technique shows how an estimation tool's product, process and resource requirements may be identified and modelled to satisfy the requirements to extend its capabilities.

3) An Integrated Product-Process-Resource-Production Cost Estimation Technique. A cost estimation model was developed that integrate design and production cost values in the graphical representation. Most cost estimation tools only concentrate on the production process cost to generate unit and volume cost, this research has identified an opportunity of integrating product, process and resource design cost with results obtained from a commercial estimation tool. The cost model developed is useful for making engineering cost analysis by changing various parameters to see its implications on the total. This technique, therefore, creates a relationship between product designers and manufacturing together by introducing cost as a key performance indicator and providing cost values that are available and visible to all.

6.4 Recommendations and Future Works

In view of the limitations of the proposed methodology (section 5.6), the author recommends the following:

- Building more complex cost equations and algorithms into the framework as the framework currently only uses time and rate for the cost model. This may include the use of other detailed indirect operational cost elements such as breakdown maintenance, human factors (absence, sick, holidays, etc.), environmental, health and safety, quality, etc. may be included in the integrated cost model to reflect a more realistic manufacturing environment. This will support the running of more demanding scenarios to stretch and test the algorithms.

- Integrating a behavioural tool such as a discrete event simulation (DES) tools with the proposed framework that graphically represents real working conditions and at the same time capable of reflecting changes in real time may support the decision making process. This will extend the framework's capabilities to the use of statistical data analysis.

- Integrating the framework with product lifecycle management (PLM) tools such as SAP PLM, Enovia, Siemens Teamcenter, etc. in order to support engineering decision making by viewing cost early during early design stages.

- Other engineering cost estimation plateforms may also be used apart from aPriori to further validate the framework.

- Simplification of the steps involved in the use of the methodologies to make it more user-friendly will be a great improvement to the framework. Currently, there are many steps to following due to the use of various tools and techniques which may mean that the methodology may only be interesting to researchers and consultants. This is because comparing the application of PPR Cost Estimation Framework with other costing estimation techniques may prove that the use of a standard or a commercial tool may be simpler to construct but its result may not be exhaustive.

- The author recommends the use of the framework on complex industrial environment to obtain cost values and comparing the results obtained with an existing cost estimation technique results. As the methodology has only been used in sheet metal fabrication application within the automotive industry, it would also be of interest to validate the methodology for estimating cost in new techniques for machining, casting, etc. Current state cost values (As-Is) and estimated cost values (To-Be) for an existing cost estimation and the proposed methodology may be obtained and compared with each other to observe flexibility and consistency of systems.

- Although the proposed methodology shows great benefits to both academic and industrial applications, the author believes that improving the cost equations of the framework to include other direct and indirect costs elements that may be key determinates of actual business process cost will be beneficial in generating better cost estimates.

6.5 Concluding Remarks

In conclusion to this research, the author is of the view that the proposed PPR Cost Estimation Framework:

1. Considers product costing from a correlation of product features with the process, resource and cost accounting data. The outcome of the research showed that the cost implication of alternative design and engineering decisions can be estimated at an early stage of the design process. This is crucial because failure to test the cost implication of an engineering decision at an early stage of the design process can result in the development of very expensive and non-competitive products and manufacturing systems.
2. Views products as being realised in a production environment which may be real or virtual. This production environment can be customised to company-based resource and cost databases to fully represent actual production system. In effect, if the behaviour of a typical production environment can be mimicked through a digital manufacturing or factory model, then based on this behaviour, feasible processes capable of meeting product requirements can be predicted within the given capacity of the modelled production system.
3. Helps to economically justify the need for product, process and resource changes. This is particularly necessary for a new product or process introduction where the cost implications of critical engineering decisions have to be fully understood. It, therefore, serves to integrate different knowledge sets needed for an early stage full digital lifecycle analysis of product manufacturing.

Despite the above capabilities of the PPR Cost Estimation Framework, the current approach is based on a 'feed-forward' mechanism which in effect predicts product manufacturing from a library of processes and resources. The 'reverse feed mechanism' brings an interesting research challenge for determining suitable products that can be realised based on process, resource and cost constraints. The reverse feed is also necessary since, typically, industries operate with finite resources and budget allocations hence products can only be realised within the scope of competence and financial capacity of the industry.

REFERENCES

A. Corallo, R. Laubacher, A. Margherita, G. Turrisi, “Enhancing product development through knowledge-based engineering (KBE): a case study in the aerospace industry”, Journal of Manufacturing Technology Management, vol. 20, 2009, pp. 1070-1083.

A.H. Van der Laan, “Knowledge based engineering support for aircraft component design”, Design of Aircraft and Rotorcraft, Faculty of Aerospace Engineering, Delft University of Technology, Delft, 2008, pp. 254.

Abdmouleh, A., Spadoni, M. & Vernadat, F. 2004. Distributed client/server architecture for CIMOSA-based enterprise components. Computers in Industry, 55, 239-253.

Abercrombie, N., Hill, S. & Turner, B. S. 1984. Dictionary of sociology, Penguin Books.

Adams, R. S., Daly, S. R., Mann, L. M. & Dall'Albad G. Being a professional: Three lenses into design thinking, acting, and being. Volume 32, Issue 6, November 2011, Pages 588-607.

Agyapong-Kodua, K. & Weston, R. 2011. Systems approach to modelling cost and value dynamics in manufacturing enterprises. International Journal of Production Research, 49, 2143-2167.

Agyapong-Kodua, K. & Weston, R. H. 2010. Systems approach to modelling cost and value dynamics in manufacturing enterprises. International Journal of Production Research, 49, 2143-2167.

Agyapong-Kodua, K. 2009. Multi-product cost and value stream modelling in support of business process analysis. PhD, Loughborough University.

Agyapong-Kodua, K. E. A., Brown, R., Darlington, R. & Ratchev, S. 2012. An integrated product–process design methodology for cost-effective product realization. International Journal of Computer Integrated Manufacturing, 25, 814-828.

Agyapong-Kodua, K., Ajaefobi, J. O. & Weston, R. H. 2009. Modelling dynamic value streams in support of process design and evaluation. International Journal of Computer Integrated Manufacturing, 22, 411-427.

Agyapong-Kodua, K., Ajaefobi, J. O., Weston, R. H. & Ratchev, S. 2012. Development of a multi-product cost and value stream modelling methodology. International Journal of Production Research, 50, 6431-6456.

Agyapong-Kodua, K., Asare, K. B. & Ceglarek, D. J. 2014. Digital Modelling Methodology for Effective Cost Assessment. Procedia CIRP, 17, 744-749.

Agyapong-Kodua, K., Brown, R., Darlington, R. & Ratchev, S. 2012c. An integrated product–process design

Agyapong-Kodua, K., Haraszkó, C. & Németh, I. 2014. Recipe-based Integrated Semantic Product, Process, Resource (PPR) Digital Modelling Methodology. Procedia CIRP, 17, 112-117.

Agyapong-Kodua, K., Wahid, B. & Weston, R. 2011. Towards the derivation of an integrated process cost-modelling technique for complex manufacturing systems. International Journal of Production Research, 49, 7361-7377.

Agyapong-Kodua, K., Wahid, B. M. & Weston, R. H. 2011. Towards the derivation of an integrated process cost modelling technique for complex manufacturing systems. International Journal of Production Research, 49, 7361-7377.

Akintoye, A. & Fitzgerald, E. 2000. A survey of current cost estimating practices in the UK. Construction Management & Economics, 18, 161-172.

Alexopoulos, K., Papakostas, N., Mourtzis, D. & Chryssolouris, G. 2011. A method for comparing flexibility performance for the lifecycle of manufacturing systems under capacity planning constraints. International Journal of Production Research, 49, 3307-3317.

Altshuller, G.S., 1999. TRIZ. The Innovation Algorithm; Systematic Innovation and Technical Creativity. Worcester, MA: Technical Innovation Centre.

AMICE 1993. CIMOSA Open Systems Architecture for CIM. Berlin: Springer Verlag.

Amue, G.J. & Adiele, K.C., 2012. New product development and consumer innovative behaviour: an empirical validation study. European Journal of Business and Social Sciences, 1(6), pp.97–109.

Andreasen M.M. Integrated Product Development, Springer Verlag, Berlin, 1987.

Angele, J. OntoBroker: Mature and approved semantic middleware.

Angele, J., Fensel, D., Landes, D., Studer, R. "Developing knowledge-based systems with MIKE", Automated Software Engineering, Vol. 5, No. 4, pp. 389–418, 1998.

aPriori. Priori For Cost Estimators [Online]. Available: http://www.apriori.com/product-cost-management-for-costestimators. html [Accessed 10/09/2016].

Arakji, R.Y. & K.R., L., 2007. 'Digital consumer networks and producer-consumer collaboration: innovation and product development in the video game industry'. Journal of Management Information Systems, 24(2), pp.195–219.

Arora, J., 2004. Introduction to Optimum Design 2nd ed., Amsterdam: Elsevier Academic Press.

Asiedu, Y., Besant, R. W. & Gu, P. 2000. Simulation-based cost estimation under economic uncertainty using kernel estimators. International Journal of Production Research, 38, 2023-2035.

AUTOFORM. AutoForm-CostEstimator [Online]. Available: http://www.autoform.com/en/products/autoform-costestimatorplus/ [Accessed 10/09/2014 2014].

Azevedo, A. & Almeida, A. 2011. Factory Templates for Digital Factories Framework. Robotics and Computer-Integrated Manufacturing, 27, 755-771.

Azevedo, A., Almeida, A., Bastos, J. & Piedade, R. 2010. Virtual factory framework: an innovative approach to support the planning and optimization of the next generation factories. Management and Control of Production and Logistics, 320-325.

Babic B, Nesic N, Miljkovic Z. A review of automated feature recognition with rule-based pattern recognition. Comput Ind 2008;59(4):321–37.

Baines, T. S., Harrison, D. K., Kay, J. M. & Hamblin, D. J. 1998. A consideration of modelling techniques that can be used to evaluate manufacturing strategies. International Journal of Advanced Manufacturing Technology, 14, 369–375.

Barnes, C.J. & Lillford, S.P., 2007. Affective design decision-making - issues and opportunities. CoDesign, 3(1), pp.135–146.

Baxter, D., Gao, J., Case, K., Harding, J., Young, B., Cochrane, S. & Dani, S. 2007. An engineering design knowledge reuse methodology using process modelling. Research in Engineering Design, 18, 37-48.

Belton, V. & Stewart, T.J., 2002a. Implementation of MCDA: Practical Issues and Insights. In Multiple Criteria Decision Analysis. Boston, MA: Springer US, pp. 261–292.

Belton, V. & Stewart, T.J., 2002b. Multiple Criteria Decision Analysis, Boston, MA: Springer US.

Beltramo, M. N. 1988. Beyond Parametrics: the role of subjectivity in cost models. Engineering Costs and Production Economics, 14, 131-136.

Bernard, F., "A short history of CATIA & Dassault Dassault Systemes, 2003. URL: http:/www.edstechnologies.com/download/history-catia.pdf

Bernus, P. & A. L. N. 1994. A Framework to Define a Generic Enterprise Reference Architecture and Methodology. Proceedings of the International Conference on Automation, Robotics and Computer Vision (ICARCV'94). Singapore.

Bernus, P. & Nemes, L. 1996. Enterprise integration-engineering tools for designing enterprises, Australia, Chapman & Hall.

Bhamra, T. et al., 1999. Integrating environmental decisions into the product development process. I. The early stages. Proceedings EcoDesign, pp.329–333.

Boehm, B. 1984. Software Engineering Economics. IEEE Trans Software Engineering 10, 7-19.

Boothroyd, G. & Reynolds, C. 1989. Approximate cost estimates for typical turned parts. Journal of Manufacturing Systems, 8, 185-193.

Borrego, M. 2007. Development of engineering education as a rigorous discipline: A study of the publication patterns of four coalitions. Journal of Engineering Education, 96, 5-18.

Bor-Tsuen Lin, Ming-Ren Chang, “A Functional-Based Stack-Up Design System for Stamping Dies”, applied mechanics and materials, vol. 110-116, 2012, pp. 1447-1457

Bouchart, F.J.-C., Blackwood, D.J. & Jowitt, P.W., 2002. Decision mapping: Understanding decision making processes. Civil Engineering and Environmental Systems, 19(3), pp.187–207.

Briand, L. C., El Emam, K., Surmann, D., Wieczorek, I. & Maxwell, K. D. An assessment and comparison of common software cost estimation modelling techniques. Proceedings of the 21st international conference on Software engineering, 1999. ACM, 313-322.

Bronsvoort, W. F. & Jansen, F. W. 1993. Feature modelling and conversion—key concepts to concurrent engineering. Computers in Industry, 21, 61-86.

Burkel, J. P. 1991. Applying CIM for competitive advantage, Society of Manufacturing Engineers.

Burton, L. 2002. Methodology and methods in mathematics education research: Where is" The Why. Researching mathematics classrooms: A critical examination of methodology, 1-10.

C.B. Chapman, M. Pinfold, “The application of a knowledge based engineering approach to the rapid design and analysis of an automotive structure”, Advances in Engineering Software, vol. 32, 2001, pp. 903–912.

C.L. Emberey, N. Milton, J.P.T.J. Berends, M.J.L. VanTooren, S.W.G. Van der Elst, B. Vermeulen, “Application of Knowledge Engineering Methodologies to Support Engineering Design Application Development in Aerospace”, 7th AIAA Aviation Technology, Integration and Operations Conference (ATIO), Belfast, Northern Ireland, 2007.

Caputo, A. C. & Pelagagge, P. M. 2008. Effects of product design on assembly lines performances: a concurrent engineering approach. Industrial Management & Data Systems, 108, 726-749.

Caputo, A. C. & Pelagagge, P. M. 2008. Parametric and neural methods for cost estimation of process vessels. International Journal of Production Economics, 112, 934-954.

Carter, D. E. & Baker, B. S. 2002. CE, concurrent engineering: the product development environment for the 1990s, Addison-Wesley Reading, MA.

Cavalieri, S., Maccarrone, P. & Pinto, R. 2004. Parametric vs. neural network models for the estimation of production costs: A case study in the automotive industry. International Journal of Production Economics, 91, 165-177.

Cayiroglu, I.: A new method for machining feature extracting of objects using 2D technical drawings. Comput. Aided Des. 41, 1008-1019 (2009).

Ceglarek D., Colledani M., Váncza J., Kim D. Y., Marine C., Kogel-Hollacher M., Mistry A., Bolognese L. 2015. Rapid deployment of remote laser welding processes in automotive assembly systems, CIRP Annals - Manufacturing Technology, Volume 64, Issue 1, 2015, Pages 389-394.

CEN/ISO 2007. Enterprise Integration- Constructs for Modelling EN ISO 19440:2007, Work item: 00310074.

Chalmeta, R. 1997. Arquitectura de referencia para la organización integrada de la empresa, Universitat Jaume I.

Chalmeta, R. 2000. Virtual transport enterprise integration. Journal of Integrated Design and Process Science, 4, 45-55.

Chen, D., Doumeingts, G. & Vernadat, F., 2008. Architectures for enterprise integration and interoperability: Past, present and future. Computers in Industry, 59(7), pp.647–659.

Cheung, J., Scanlan, J. & Wiseall, S. 2009. An aerospace component cost modelling study for value driven design.

Chryssolouris, G., Mavrikios, D., Papakostas, N., Mourtzis, D., Michalos, G. & Georgoulias, K. 2009. Digital manufacturing: history, perspectives, and outlook.

Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 223, 451-462.

Chung, S. & Synder, C. 1999. ERP initiation-a historical perspective. AMCIS 1999 Proceedings, 76.

Chwastyk, P. & Kołosowski, M., 2014. Estimating the Cost of the New Product in

Colledani, M., Pedrielli, G., Terkaj, W. & Urgo, M. 2013. Integrated Virtual Platform for Manufacturing Systems Design. Procedia CIRP, 7, 425-430.

Colledani, M., Tolio, T., Fischer, A., Iung, B., Lanza, G., Schmitt, R. & Váncza, J. 2014. Design and management of manufacturing systems for production quality. CIRP Annals-Manufacturing Technology, 63, 773-796.

Collis, J. & Hussey, R. 2003. Business research: A practical guide for postgraduate and undergraduate students. New York: Palgrave Macmillan.

Collopy, P. & Curran, R. The challenge of modelling cost: the problem. Proceedings of the 1st International Conference on Innovation and Integration in Aerospace Sciences, Belfast, United Kingdom, 2005.

Conteh, C., 2013. Strategic Inter-Organizational Cooperation in Complex Environments. Public Management Review, 15(4), pp.501–521.

Cooper, R. & Kaplan, R. S. 1987. How cost accounting systematically distorts product costs. Accounting and management: Field study perspectives, 204-228.

Cooper, R. & Kaplan, R. S. 1999. The design of cost management systems: text and cases, Prentice Hall.

Cooper, R. 1988. The Rise of Activity Based Costing Part One: What is an Activity Based Cost System?

Cooper, S., Fan, I., Li, G., "Achieving competitive through knowledge-based engineering", Cranfield University, 2001.

Crawford, G. 1985. The analysis of subjective judgement matrices. The Rand Corporation.

Curran, R., Raghunathan, S. & Price, M. 2004. Review of aerospace engineering cost modelling: The genetic causal approach. Progress in Aerospace Sciences, 40, 487-534.

Curran, R., Raghunathan, S. & Price, M. 2004. Review of aerospace engineering cost modelling: The genetic causal approach. Progress in Aerospace Sciences, 40, 487-534.

Curran, R., Watson, P., Cowan, S., Mahwinney, J. & Raghunathan, S. 2003. Development of an aircraft cost estimating model for program cost rationalisation. Montreal Proceedings of the Canadian Aeronautics and Space Institute (CASI).

D. Arthur, A.Ramsay & R.Samanta 2004. <A Budgetary Analysis of NASA's New Vision for Space Exploration.pdf>. Congress Of The United States Congressional Budget Office.

D. Baxter, J. Gao, K. Case, J. Harding, B. Young, S. Cochrane, S. Dani, “An engineering design knowledge reuse methodology using process modelling”, Research in Engineering Design, vol. 18, 2007, pp. 37–48.

D'addona, D. & Teti, R. 2011. Multi-agent tool management in the manufacturing of aircraft engines. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 225, 62-73.

Dall'alba, G. 2011. Being a professional: Three lenses into design thinking, acting, and being. Design Studies, 32, 588-607.

Daly, S. R., Yilmaz, S., Christian, J. L., Seifert, C. M. & Gonzalez, R. 2012. Design heuristics in engineering concept generation. Journal of Engineering Education, 101, 601-629.

Dassault Systemes. “Dassault Systemes acquires KTI”, Systems Website, 2002. URL: http://www.3ds.com/news-events/press-room/release/796/1/

David Ben-Arieh, L. Q. 2003. <Activity-based cost management for design and development stage_Ben-Arieh.pdf>. Int. J. Production Economics, 83, 169–183.

David, G., Herve, J. & Cahill, D. 2003. Cost Engineering for Cost Effective Programmes. August 2003: Cost Analysis Division, ESA Directorate for Industrial Matters and Technology Programmes, ESTEC, Noordwijk, The Netherlands.

Davies, G. 2012. Chapter 6 - Component assembly: materials joining technology. In: Davies, G. (ed.) Materials for Automobile Bodies. Oxford: Butterworth-Heinemann.

De Hoog, R., B. Benus, C. Metselaar, M. Vogler, and W. Menezes: Organisation Model: Model Definition Document. ESPRIT Project P5248 KADS-II, report KADS-II/M6/UvA/041/3.0, University of Amsterdam, 1994.

Decker, S.; Erdman, M.; Studer, R. A Unifying View on Business Process Modelling And Knowledge Engineering. In Proceedings of the 10th Kew (Kaw96), Banff, Canada. 1996.

Desmond, C., Brubaker, K.A. & Ellner, A.L., 2013. Decision-making strategies: ignored to the detriment of healthcare training and delivery? Health psychology and behavioral medicine, 1(1), pp.59–70.

Dietz, J. L. 2006. What is Enterprise Ontology? Springer.

Dongkon Lee, "Knowledge-based system for safety control of damaged ship", Knowledge-Based Systems, vol. 19, 2006, pp. 187–191.

Duggan Manufacturing, "New Process Development Facility for Remote Fiber Laser Welding," 2010. http://weldingdesign.com/equipment-automation/news/pr ocess development-facility-remote-fiber-laser-0313

Dunn, J.A. et al., 2013. Liminality and decision making for upper limb surgery in tetraplegia: a grounded theory. Disability and Rehabilitation, 35(15), pp.1293–1301.

Duverlie, P. & Castelain, J. M. 1999. Cost Estimation During Design Step: Parametric Method versus Case Based Reasoning Method. The International Journal of Advanced Manufacturing Technology, 15, 895-906.

Dwarakanath, S. & Wallace, K.M., 1995. Decision-making in Engineering Design: Observations from Design Experiments. Journal of Engineering Design, 6(3), pp.191–206.

E. Jayakiran Reddy, C. N. V. Sridhar and V. Pandu Rangadu. Knowledge Based Engineering: Notion, Approaches and Future Trends. American Journal of Intelligent Systems 2015, 5(1): 1-17

Eden, C. & Radford, J., 1990. Tackling Strategic Problems: the Role of Group Decision Support. London, Sage.

Elmaraghy, H. A. 2005. Flexible and reconfigurable manufacturing systems paradigms. International journal of flexible manufacturing systems, 17, 261-276.

Eppinger, S., Whitney, D., Smith, R. & Gebala, D. 1994. A model-based method for organizing tasks in product development. Research in Engineering Design, 6, 1-13.

Erdős, G., Kemény, Z., Kovács, A. & Váncza, J. 2013. Planning of Remote Laser Welding Processes. Procedia CIRP, 7, 222-227.

Factories. Digital Factories Theme., in, pp. http://www.rlwnavigator.eu/acknowledgements/.

Farsi, M., Arezoo, B.: Feature Recognition and AND Design Advisory System for Sheet Metal Components. Presented at the International Advanced Technologies Symposium, Turkey Mayo 13 (2009).

Fayoumi, A., 2016. Ecosystem-inspired enterprise modelling framework for collaborative and networked manufacturing systems. Computers in Industry, 80, pp.54–68.

Flyvbjerg, B. 2001. Making social science matter: Why social inquiry fails and how it can succeed again, Cambridge university press. For aerospace composite manufacturing. Advanced Manufacturing: Polymer & Composites Science.

Force, I.-I. T. IFIP–IFAC Task Force on Architectures for Enterprise Integration 1999. Version 1.6.3.

Foussier, P. M. M. 2006. From Product description to Cost: A Practical Approach, Berlin, Springer; Vol 1: The parametric approach.

Fuchs, C., Prandelli, E. & Schreier, M., 2010. 'The psychological effects of empowerment strategies on consumers' product demand'. Journal of Marketing, 74(1), pp.65–79.

G. La Rocca, M.J.L. Van Tooren, “Enabling distributed multi-disciplinary design of complex products: a knowledge based engineering approach”, Journal of Design Research, vol. 5, 2007, pp. 333–352.

Georgia N. Koini, Sotirios S. Sarakinos, Ioannis K. Nikolos, “A software tool for parametric design of turbomachinery blades”, Advances in Engineering Software, vol. 40, 2009, pp. 41–51.

Gershwin, S. 1994. Manufacturing systems engineering, 1994. PTR Prentice Hall.

Gholipour, R. & Rouzbehani, K. eds., 2016. Social, Economic, and Political Perspectives on Public Health Policy-Making, IGI Global.

Ginn, D. & Zairi, M., 2005. Best Practice QFD Application: An Internal/External Benchmarking Approach Based on Ford Motors’ Experience.”. International Journal of Quality & Reliability Management, 22, pp.38–58.

Goncalves-Coelho, A.M. & Mourao, A.J.F., 2007. Axiomatic Design as Support for Decision-Making in a Design for Manufacturing Context: A Case Study". International Journal of Production Economics, 109, pp.81–89.

Gray, A. R., Macdonell, S. G. & Shepperd, M. J. Factors systematically associated with errors in subjective estimates of software development effort: the stability of expert judgment. Software Metrics Symposium, 1999. Proceedings. Sixth International, 1999. IEEE, 216-227.

Group, O. 2005. TOGAF (The Open Group Architectural Framework) Version 8.1 “Enterprise Edition”.

Grupp M., Seefeld T. and Vollertsen F., “Laser Beam Welding with Scanner,” Proceedings of the Second International WLT-Conference on Lasers in Manufacturing, Munich, 2003, pp. 211-222.

Gunasekaran, A. & Sarhadi, M. 1998. Implementation of activity-based costing in manufacturing. International journal of production economics, 56, 231-242.

Hall, N. A. and Delille, S. 2011. Cost Estimation Challenges and Uncertainties Confronting Oil and Gas Companies. *2011 AACE International Transactions*.

H.Z. Yang, J.F. Chen, N. Ma, D.Y. Wang, "Implementation of knowledge-based engineering methodology in ship structural design", Computer-Aided Design, vol. 44, 2012, pp. 196–202.

Hajare, A. Parametric costing- steel wire mill. Wire Association International 68 the Annual Convention, 1998. 172-178.

Hamann, T. & Vernadat, F. 1992. The Intra-cell layout problem in automated manufacturing systems.

Han J, Pratt M, Regli WC. Manufacturing feature recognition from solid models: a status report. IEEE Trans Robot Autom 2000; 16(6):782–96.

Hao, M., Osman, K., Boomer, D. & Newton, C. 1996. Developments in characterization of resistance spot welding of aluminium. Welding Journal-Including Welding Research Supplement, 75, 1-4.

Heragu, S. S. & Kusiak, A. 1988. Machine layout problem in flexible manufacturing systems. Operations Research, 36, 258-268.

Herrmann, J., 2015. Engineering Decision Making and Risk Management, Hoboken, New Jersey: John Wiley & Sons, Inc Hoboken, New Jersey.

Hibino, H., Inukai, T. & Fukuda, Y. 2006. Efficient manufacturing system implementation based on combination between real and virtual factory. International Journal of Production Research, 44, 3897-3915.

Hilton, M.D. & P, M.J., 2016. Enterprise Integration of Engineering Systems for Defence Related Projects. Transdisciplinary Engineering: Crossing Boundaries.

Hoepfl, M. C. 1997. Choosing qualitative research: A primer for technology education researchers.

Holt, R. & Barnes, C. 2010. Towards an integrated approach to "Design for X": an agenda for decision-based DFX research. Research in Engineering Design, 21, 123-136.

Horlick-Jones, T. et al., 2001. Decision support for organizational risk management by problem structuring. Available at: http://www.tandfonline.com/action/journalInformation?journalCode=chrs20 [Accessed January 17, 2017].

Howard C. C, The Virtual Engineer, 21st Century Project Development – Society Of Manufacturing Engineers 1998, ISBN: 0-87263-491-4

Hueber, C. K. H. R. S. 2016. Review of cost estimation: methods and models

Hughes, R.T., 1996. Expert judgement as an estimating method. Information and Software Technology, 38(2), pp.67–75.

Intelligent, C. ltd, 2009. Doing Enterprise Architecture: Enabling the agile institution. Technology & Standards Watch Early Adopter Study JISC.

Internal report of RLW project 2014. Remote Laser Welding (RLW) System Navigator for Eco and Resilient Automotive

International Society of Parametric Analysts

ISPA 2008 Parametric Estimating Handbook Fourth Edition

J. Owen, STEP – An Information, Information Geometers, (1993).

J. Scanlan, A. Rao, C. Bru, P. Hale & Marsh, R. 2006. DATUM project: cost estimating environment for support of aerospace design decision making. Journal of Aircraft, 43, 1022–1028.

J. Ravat, J.W. Nazemetz, Introduction to STEP, CATT Research program

J.W. Choi, D. Kelly, J. Raju, "A knowledge-based engineering tool to estimate cost and weight of composite aerospace structures at the conceptual stage of the design process", Aircraft Engineering and Aerospace Technology, vol. 79, 2007, pp. 459–468.

J.W. Choi, D. Kelly, J. Raju, C. Reidsema, "Knowledge-based engineering system to estimate manufacturing cost for composite structures", Journal of Aircraft vol. 42, 2005, pp. 1396–1402.

Jain, S., Choong, N. F., Aye, K. M. & Luo, M. 2001. Virtual factory: an integrated approach to manufacturing systems modeling. International Journal of Operations & Production Management, 21, 594-608.

Jamil A. Khan, L. X. Y.-J. C. K. B. 2000. Numerical Simulation of Resistance Spot Welding Process. Numerical Heat Transfer, Part A: Applications, 37, 425-446.

Jarratt, T. A. W., Eckert, C. M., Caldwell, N. H. M. & Clarkson, P. J. 2011. Engineering change: an overview and perspective on the literature. Research in Engineering Design, 22, 103-124.

Jawahar Lal. (2009). Cost Accounting. New Delhi: Tata Mcgaw-Hill Publishing.

V. K. Saxena & C. D. Vashist. (2010). Advanced Management Accounting. New Delhi. Sultan Chand & Sons.

Jin, Y., Curran, R., Burke, R. & Welch, B. 2011. An integration methodology for automated recurring cost prediction using digital manufacturing technology. International Journal of Computer Integrated Manufacturing, 25, 326-339.

Jin, Y., Curran, R., Burke, R. & Welch, B. 2012. An integration methodology for automated recurring cost prediction using digital manufacturing technology. International Journal of Computer Integrated Manufacturing, 25, 326-339.

Jin-Woo Choi, "Architecture of a knowledge based engineering system for weight and cost estimation for a composite airplane structures", Expert Systems with Applications, vol. 36, 2009, pp. 10828–10836.

Johnson, H. & Kaplan, R. 1987. Relevance Lost – The Rise and Fall of Management Accounting, Boston, MA, Harvard Business School Press.

Jon, S. & Greene, R. 2003. Sociology and you. Ohio: Glencoe McGraw-Hill.

Julien, B. et al., 1990. Civil Engineering Systems A fuzzy screening model for multiple attribute decision-making A fuzzy screening model for multiple attribute decision-making.

Kaluza, A. et al., 2016. Analyzing Decision-making in Automotive Design towards Life Cycle Engineering for Hybrid Lightweight Components. Procedia {CIRP}, 50, pp.825–830.

Kassem, M., Dawood, N. & Mitchell, D., 2011. A Structured Methodology For Enterprise Modeling: A Case Study For Modeling The Operation Of A British Organization. Journal of Information Technology in Construction, 16, pp.381–410.

Kazmer, D. & Roser, C. 1999. Evaluation of Product and Process Design Robustness. Research in Engineering Design, 11, 20-30.

Keller, S., Collopy, P. & Componation, P. 2014. What is wrong with space system cost models? A survey and assessment of cost estimating approaches. Acta Astronautica, 93, 345-351.

Khataie, A.H. & Bulgak, A. a., 2013. A cost of quality decision support model for lean manufacturing: activity-based costing application. International Journal of Quality & Reliability Management, 30(7), pp.751–764.

Kim, J.-G. & Kim, Y.-D. 2000. Layout planning for facilities with fixed shapes and input and output points. International Journal of Production Research, 38, 4635-4653.

Kirsch, U., 1981. Optimum Structural Design, New York: McGraw-Hill Book Company.

Knutson, S. "Knowledge based engineering Knutson Website, 2003. URL: http://www.stanleyknutson.com/kbe-history.html

Koonce, D., R. Judd, et al. (2003). "A hierarchical cost estimation tool." Computers in Industry **50**(293-302).

Kosanke, K. 1995. CIMOSA—overview and status. Computers in industry, 27, 101-109.

Kosanke, K. 1996. Process Oriented Presentation of Modelling Methodologies. Proceedings of the IFIP TC5 Working Conference on Models and Methodologies for Enterprise Integration, 45-55.

Kosanke, K., Vernadat, F. & Zelm, M. 1999. CIMOSA: enterprise engineering and integration. Computers in industry, 40, 83-97.

Kothari, C. R. 2004. Research methodology: methods and techniques, New Age International.

Kotler, P. & Armstrong, G., 2012. the principles of marketing 14th ed. E. Svendsen, ed., Prentice Hall.

Langmaak, S., Wiseall, S., Bru, C., Adkins, R., Scanlan, J. & Sóbester, A. 2013. An activity-based-parametric hybrid cost model to estimate the unit cost of a novel gas turbine component. International Journal of Production Economics, 142, 74-88.

Layer, A., Brine, E. T., Van Houten, F., Kals, H. & Haasis, S. 2002a. Recent and future trends in cost estimation. International Journal of Computer Integrated Manufacturing, 15, 499-510.

Lê, L.-S. & Wegmann, A. 2013. Hierarchy-oriented modeling of enterprise architecture using reference-model of open distributed processing. Computer Standards & Interfaces, 35, 277-293.

Lee, J., Han, S. & Yang, J. 2011. Construction of a computer-simulated mixed reality environment for virtual factory layout planning. Computers in Industry, 62, 86-98.

Lewis, R. J. 1993. Activity-based costing for marketing and manufacturing, Quorum Books.

Li, H. & Williams, T. J. 1994. A formalization and extension of the Purdue enterprise reference architecture and the Purdue methodology. Also published as Technical Report 158Purdue University, West Lafayette, IN, USA.

Loucopoulos, P. & Kavakli, E., 1995. Enterprise modelling and the teleological approach to requirements engineering. Int. J. Cooperative Inf. Syst., 4, pp.45–79.

Maguire, S., 2002. Discourse and adoption of innovations: A study of HIV/AIDS treatments. Health Care Management Review, 27, pp.74–89.

Mäntylä M, Nau D, Shah J. Challenges in feature-based manufacturing research. Commun ACM 1996;39(2):77–85.

Marinov, V. 2000. What Virtual Manufacturing is? Part I: Definition.

Mark, S.F. & Gruninger, M., 1998. Enterprise Modeling. American Association for Artificial Intelligence.

Maropoulos, P. G. & Ceglarek, D. 2010. Design verification and validation in product lifecycle. CIRP Annals-Manufacturing Technology, 59, 740-759.

Maropoulos, P. G. & Ceglarek, D. 2010. Design verification and validation in product lifecycle. CIRP Annals - Manufacturing Technology, 59, 740-759.

Maskell, B. H. 1991. Performance Measurement for World Class Manufacturing - A model for American Companies, Cambridge, Massachusetts, Productivity Press, Inc.

Matsuda, M., Kashiwase, K. & Sudo, Y. 2012. Agent Oriented Construction of a Digital Factory for Validation of a Production Scenario. Procedia CIRP, 3, 115-120.

Matta, A., Semeraro, Q. & Tolio, T. 2005. A framework for long term capacity decisions in AMSs. Design of advanced manufacturing systems. Springer.

Matthew G. and Chriatine G. Remote laser welding boosts production of new Ford Mustang: technology enables efficiencies in time, space, and productivity. Industrial Laser Solutions for Manufacturing. Annual Economic Review. January/February 2017.

Mingers, J. & Gill, A., 1997. Multimethodology: the Theory and Practice of Combining Management Science Methodologies, (Chichester, Wiley.

Moen, P. & Erickson, M.A., 2001. Chapter 3 Decision-Making and Satisfaction with a Continuing Care Retirement Community. Journal of Housing For the Elderly, 14(1-2), pp.53–69.

Mohanbir Sawhney, Gianmario Verona, Emanuela Prandelli, “Collaborating to create: the internet as a platform for customer engagement in product innovation”, Journal of interactive marketing, vol. 19, 2005, pp.4-17

MOKA consortium, “Managing engineering knowledge”, Professional Engineering Publishing Limited, London, 2001

Molina, A., Al-Ashaab, A., Ellis, T. A., Young, R. M. & Bell, R. 1995. A review of computer-aided Simultaneous Engineering systems. Research in Engineering Design, 7, 38-63.

Mori, K., Tarui, T., Hasegawa, T. & Yoshikawa, N. 2010. Remote laser welding applications for car bodies. Welding International, 24, 758-763.

Mümtaz İpek, İhsan H. Selvi, Fehim Findik, Orhan Torkul, I.H. Cedimoğlu, "An expert system based material selection approach to manufacturing", Materials & Design, vol. 47, 2013, pp. 331–340.

Negahban, A. & Smith, J. S. 2014. Simulation for manufacturing system design and operation: Literature review and analysis. Journal of Manufacturing Systems, 33, 241-261.

Niazi, A., Dai, J. S., Balabani, S. & Seneviratne, L. 2006. Product Cost Estimation: Technique Classification and Methodology Review. Journal of Manufacturing Science and Engineering, 128, 563.

Nicolas Gardan, Yvon Gardan, "An application of knowledge based modelling using scripts", Expert Systems with Applications, vol. 25, 2003, pp. 555–568.

Nied, H. 1984. The finite element modeling of the resistance spot welding process. Weld. J., 63, 123.

Nikolaos, T., Erik, J.H. & Susan, H., 2004. "Navigating the New Product Development Process. Journal of Industrial Marketing Management, 33(619-626).

Noran, O. 2005. A systematic evaluation of the C4ISR AF using ISO15704 Annex A (GERAM). Computers in Industry, 56, 407-427.

Noran, O. 2012. Achieving a sustainable interoperability of standards. Annual Reviews in Control, 36, 327-337.

Ostwald, P. F. 1992. Engineering Cost Estimating, Prentice Hall.

Ou-Yang, C. & Lin, T. 1997. Developing an integrated framework for feature-based early manufacturing cost estimation. The international journal of advanced manufacturing technology, 13, 618-629.

Özbayrak, M., Akgün, M. & Türker, A. 2004. Activity-based cost estimation in a push/pull advanced manufacturing system. International journal of production economics, 87, 49-65.

P. Bermell-Garcia, I.S. Fan, “A KBE System for the Design of Wind Tunnel Models Using Reusable Knowledge Components”, International Congress on Project Engineering, Barcelona, 2002.

Pace, S., Et Al 1995. The global positioning system. RAND CORPORATION.

Papalambros, P.Y. & Wilde, D., 2000. Principles of Optimal Design 2nd ed., Cambridge: Cambridge University Press.

Park, C. S. & Kim, G.-T. 1995. An economic evaluation model for advanced manufacturing systems using activity-based costing. Journal of Manufacturing Systems, 14, 439-451.

Park, J. & Simpson, T. W. An activity-based costing method for product family design in the early stages of development. ASME 2005 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, 2005. American Society of Mechanical Engineers, 959-968.

Pedrazzoli, P., Sacco, M., Jönsson, A. & Boër, C. 2007. Virtual Factory Framework: Key Enabler For Future Manufacturing. In: CUNHA, P. & MAROPOULOS, P. (eds.) Digital Enterprise Technology. Springer US.

Peijun Wang, Robert Bjärnemo, Damien Motte, "Development of a web-based customer-oriented interactive virtual environment for mobile phone design", Proceedings of DETC'03, ASME 2003 Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Chicago, Illinois, USA

Polmear, I., Pons, G., Barbaux, Y., Octor, H., Sanchez, C., Morton, A., Borbidge, W. & Rogers, S. 1999. After concorde: evaluation of creep resistant Al–Cu–Mg–Ag alloys. Materials Science and Technology, 15, 861-868.

Popov, V., 2010. The use of a virtual building design and construction model for developing an effective project concept in 5D environment. Automation in Construction. Available at: doi: 10.1016/j.auction.2009.12.005.

Prakash, Eduardo Izquierdo, L. & Ceglarek, D. 2009. Functional process adjustments to reduce No-Fault-Found product failures in service caused by in-tolerance faults. CIRP Annals - Manufacturing Technology, 58, 37-40.

Pugh, S., 1981. ‘Concept Selection: A Method That Works.’ In Proceedings of the International Conference on Engineering Design ICED, Rome, Italy.

Qian, L. & Ben-Arieh, D., 2008. Parametric cost estimation based on activity-based costing: A case study for design and development of rotational parts. International Journal of Production Economics, 113(2), pp.805–818.

Quade, E.S., 1967. Analysis For Military Decisions, Amsterdam, North-Holland.

Quintino, L. et al., 2012. Laser Welding of Structural Aluminium. Adv Struct Mater, 8, pp.33–57.

Radhakrishnan, R., Amsalu, A., Kamran, M., Nnaji, B.O.: Design rule checker for sheet metal components using medial axis transformation and geometric reasoning. Journal of Manufacturing Systems. 15, 179-189 (1996).

Ramasubbu, Narayanasamy and BALAN, Rajesh Krishna. Globally Distributed Software Development Project Performance: An Empirical Analysis. The 6th Joint Meeting of the European Software Engineering Conference and the ACM SIGSOFT Symposium on the Foundations of Software Engineering: Dubrovnik, Croatia, September 3-7, 2007. 125-134.

Rangel, L.A.D., Gomes, L.F.A.M. & Moreira, R.A., 2009. Decision theory with multiple criteria: an aplication of ELECTRE IV and TODIM to SEBRAE/RJ. Pesquisa Operacional, 29(3), pp.577–590.

Ravindran, A., Ragsdell, K.M. & Reklaitis, G. V, 2006. Engineering Optimization: Methods and Applications, Hoboken, New Jersey, John Wiley & Sons.

Ren´e Berndt, Sven Havemann, Dieter Fellner, “3D Modeling in a Web Browser to Formulate Content-Based 3D Queries”, Proceedings of the 14th International Conference on 3D Web Technology, 2009, pp.111-118.

Renzi, C., Leali, F. & Di Angelo, L., 2017. A review on decision-making methods in engineering design for the automotive industry. Journal of Engineering Design, pp.1–26.

Roberts, C. & Hermosillo, E. 2000. An automated machining cost estimator. Journal of Engineering Valuation and Cost Analysis, 3, 27-42.

ROSEN, D. 1993. Feature-based design: Four hypotheses for future CAD systems. Research in Engineering Design, 5, 125-139.

Rouzbehani, K., 2016. Problem-structuring methods: collaborative action with an application to the healthcare sector in Iran. Asia Pacific Journal of Public Administration, 38(4), pp.281–288.

Roy, R. & Palacio, A. Cost estimating and risk analysis in manufacturing processes. Proceedings of MATADOR 2000 Conference, 2000 Manchester. UMIST, 177-182.

Roy, R. 2003. Cost engineering: why, what and how?

Rush, C. & Roy, R. 2000. Analysis of cost estimating processes used within a concurrent engineering environment throughout a product life cycle. 7th ISPE Int. Conference on Concurrent Engineering: Research and Applications. Lyon, France: Technomic, Pennsylvania, USA.

Rush, C. & Roy, R. 2001. Capturing quantitative & qualitative knowledge for cost modelling within a CE environment. In: ISPE International Conference on Concurrent Engineering: Research and Applications. Anaheim: CETEAM.

Rush, C. & Roy, R. 2001b. Expert judgement in cost estimating: modelling the reasoning process. Concurrent Engineering Research Application (CERA), 9.

Rush, C. & Roy, R. 2001c. Expert judgement in cost estimating: modelling the reasoning process. Concurrent Engineering Research Application, 9, 271-284.

Rush, C. & Roy, R. Analysis of cost estimating processes used within a concurrent engineering environment throughout a product life cycle. 7th ISPE International Conference on Concurrent Engineering: Research and Applications, Lyon, France, July 17th-20th, Technomic Inc., Pennsylvania USA, 2000. 58-67.

Rush, C. & Roy, R., 2001. Expert Judgement in Cost Estimating: Modelling the Reasoning Process. Concurrent Engineering, 9(4), pp.271–284.

Ryszard Arendt, Ewa van Uden, “A decision-making module for aiding ship system automation design: A knowledge-based approach”, Expert Systems with Applications, vol. 38, 2011, pp. 410–416.

S. Ammar-Khodja, N. Perry, A. Bernard, “Processing knowledge to support knowledge-based engineering systems specification”, Concurrent Engineering-Research and Applications, vol. 16, 2008, pp. 89–101.

S. Ammar-Khodja, N. Perry, A. Bernard, “Processing knowledge to support knowledge-based engineering systems specification”, Concurrent Engineering-Research and Applications, vol. 16, 2008, pp. 89–101.

S. Cooper, I.S. Fan, G. Li, “Achieving Competitive Advantage Through Knowledge Based Engineering – A Best Practice Guide”, Cranfield University, Bedford, UK, 2001.

S. Kumar, R. Singh, “A short note on an intelligent system for selection of materials for progressive die components”, Journal of Materials Processing Technology, vol. 182, 2007, pp. 456–461.

S. Kumar, R. Singh, “An automated design system for progressive die”, Expert Systems with Applications, vol. 38, 2011, pp. 4482–4489.

S.M. Sapuan, “A knowledge-based system for materials selection in mechanical engineering design”, Materials and Design, vol. 22, 2001, pp. 687-695

saaty, T. & Vargas, L., 2006. Decision Making With The Analytic Network Process Economic, Political, Social and Technological Applications with Benefits, Opportunities, Costs and Risks, Springer Science+Business Media, LLC.

Saaty, T., 1980. The Analytic Hierarchy Process, New York: McGraw-Hill Book Company.

Sandberg M., Knowledge Based Engineering - In Product Development, Technical Report, Lulea University of Technology, 2003:05, ISSN 1402-1536

Sandmeier, P., Morrison, P.D. & O, G., 2010. 'Integrating customers in product innovation: lessons from industrial development contractors and in-house contractors in rapidly changing customer markets'. Journal of Product Innovation Management, 12(3), pp.200–213.

Santos, I. O., Zhang, W., Gonçalves, V. M., Bay, N. & Martins, P. A. F. 2004. Weld bonding of stainless steel. International Journal of Machine Tools and Manufacture, 44, 1431-1439.

Saracoglu, B.O., 2015. An Experimental Research Study on the Solution of a Private Small Hydropower Plant Investments Selection Problem by ELECTRE III/IV, Shannon's Entropy, and Saaty's Subjective Criteria Weighting. Advances in Decision Sciences, 2015, pp.1–20.

Scanlan, J., Hill, T., Marsh, R., Bru, C., Dunkley, M. & Cleevely, P. 2002. Cost modelling for aircraft design optimization. Journal of Engineering Design, 13, 261-269.

Schekkerman, J. 2004. Enterprise Architecture Scorecard, Version 2.1. Institute for Enterprise Architecture Developments, the Netherlands.

Schlueter, H. 2007. <Laser Beam Welding_Benefits, Strategies, and Applications.pdf>. WELDING JOURNAL.

Schreiber, A. Th Schreiber, B. Wielinga, R. de Hoog, H. Akkermans, and W. van de Velde: CommonKADS: A Comprehensive Methodology for KBS Development. IEEE Expert, December 1994, 28-37

Seo, K. K., Park, J. H., Jang, D. S. & Wallace, D. 2002. Prediction of the life cycle cost using statistical and artificial neural network methods in conceptual product design. International Journal of Computer Integrated Manufacturing, 15, 541-554.

Sérgio Santos Jr, P., Almeida, J. P. A. & Guizzardi, G. 2013. An ontology-based analysis and semantics for organizational structure modeling in the ARIS method. Information Systems, 38, 690-708.

Shailendra Kumara, Rajender Singh, "An intelligent system for automatic modeling of progressive die", Journal of Materials Processing Technology, vol. 194, 2007, pp. 176–183.

Shehab, E. & Abdalla, H. 2001. Manufacturing cost modelling for concurrent product development. Robotics and Computer-Integrated Manufacturing, 17, 341-353.

Shehab, E. & Abdalla, H. 2002. A design to cost system for innovative product development. Proceedings of Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 216, 999-1019.

Shen, H. et al., 2004. Integration of business modelling methods for enterprise information system analysis and user requirements gathering. Computers in Industry, 54(3), pp.307–323.

Shepperd, M. & Cartwright, M. 2001. Predicting with sparse data. Software Engineering, IEEE Transactions on, 27, 987-998.

Shrouti, C., Franciosa, P. & Ceglarek, D. 2013. Root Cause Analysis of Product Service Failure Using Computer Experimentation Technique. Procedia CIRP, 11, 44-49.

Shtub, A. and Y. Zimerman (1993). "A neural-network-based approach for estimating the cost of assembly systems." International Journal of Production Economics., 32(2): 189–208.

Smith, A. E. & Mason, A. K. 1997. Cost estimation predictive modeling: Regression versus neural network. The Engineering Economist, 42, 137-161.

Son, Y. K. 1991. A Cost Estimation Model For Advanced Manufacturing Systems. International Journal of Production Research, 29, 441-452.

Stauffer, L.A. & Ullman, D.G., 1991. Fundamental Processes of Mechanical Designers Based on Empirical Data. Journal of Engineering Design, 2(2), pp.113–125.

Stevenson, M. & Vanharanta, M., 2015. The effects of managerial decision making behaviour and order book size on workload control system implementation in Make-To-Order companies. Production Planning & Control, 26(2), pp.97–115.

Stewart, R. D. 1991. Cost Estimating, Wiley.

Studer, R., Benjamins, V., R., Fensel, D. “Knowledge engineering: principles and methods”, Data & Knowledge Engineering, Vol. 25, Issues 1-2, pp. 161–197, 1998.

Talay, M.B., Calantone, R.J. & Voorhees, C.M., 2014. Coevolutionary Dynamics of Automotive Competition: Product Innovation, Change, and Marketplace Survival. Journal of Product Innovation Management, 31(1), pp.61–78.

Tammineni, S. V., Rao, A. R., Scanlan, J. P., Reed, P. A. S. & Keane, A. J. 2009. A knowledge-based system for cost modelling of aircraft gas turbines. Journal of Engineering Design, 20, 289-305.

Tang, A., Han, J. & Chen, P. A comparative analysis of architecture frameworks. Software Engineering Conference, 2004. 11th Asia-Pacific, 30 Nov.-3 Dec. 2004 2004. 640-647.

Terkaj, W., Danza, L., Devitofrancesco, A., Gagliardo, S., Ghellere, M., Giannini, F., Monti, M., Pedrielli, G., Sacco, M. & Salamone, F. 2014. A Semantic Framework for Sustainable Factories. Procedia CIRP, 17, 547-552.

TOGAF 2015. TOGAF®, an Open Group standard https://www.opengroup.org/togaf/. 2015

TOGAF, 2011. Open Group Standard TOGAF® Version 9.1,

Tolio, T., Sacco, M., Terkaj, W. & Urgo, M. 2013. Virtual Factory: An Integrated Framework for Manufacturing Systems Design and Analysis. Procedia CIRP, 7, 25-30.

Tornberg, K., Jämsen, M. & Paranko, J. 2002. Activity-based costing and process modeling for cost-conscious product design: A case study in a manufacturing company. International Journal of Production Economics, 79, 75-82.

Tseng, Y.-J. & Jiang, B. 2000. Evaluating multiple feature-based machining methods using an activity-based cost analysis model. The International Journal of Advanced Manufacturing Technology, 16, 617-623.

Tsoukantas, G., Stournaras, A. & Chryssolouris, G. 2008. Experimental investigation of remote welding with CO2 and Nd:YAG laser-based systems. Journal of Laser Applications, 20, 50-58.

Tuckman, B. W. & Harper, B. E. 2012. Conducting educational research, Rowman & Littlefield Publishers.

Turney, P. B. 1991. Common cents: The ABC performance breakthrough: How to succeed with activity-based costing, Cost Technology Hillsboro, OR.

Turney, P. B. 1996. Activity based costing. The Performance Breakthrough. London: CLA.

Ullman, D.G., 2001. Robust decision-making for engineering design. Journal of Engineering Design, 12(1), pp.3–13.

Ulrich, K.T. & Eppinger, S.D., 2008. Product Design and Development. 4th ed. Singapore: Irwin McGraw-Hill.

Um, J. & Stroud, I. A. 2013. Total Energy Estimation Model for Remote Laser Welding Process. Procedia CIRP, 7, 658-663.

Um, J. & Stroud, I. A. 2013. Total Energy Estimation Model for Remote Laser Welding Process. Procedia CIRP, 7, 658-663.

V. Naranje, S. Kumar, “A knowledge based system for automated design of deep drawing die for axisymmetric parts”, Expert Systems with Applications, vol. 41, 2014, pp. 1419–1431

V. Naranje, S. Kumar, “A Knowledge Based System for Selection of Components of Deep Drawing Die”, American Journal of Intelligent Systems, vol. 2, 2012, pp. 1-11

Veeramani, D. & Joshi, P. 1997. Methodologies for rapid and effective response to requests for quotation (RFQs). IIE Transactions, 29, 825-838.

Vergidis, K., Tiwari, A. & B., M., 2008. Business Process Analysis and Optimization: Beyond Reengineering. IEEE Transactions on Systems, Man and Cybernetics, Part C: Applications and Reviews, 38.

Verhagen, W. J. C., Garcia, P. B., Mariot, P., Cotton, J.-P., Ruiz, D., Redon, R. & Curran, R. 2010. Knowledge based cost modelling of composite wing structures. International Journal of Computer Integrated Manufacturing, 25, 368-383.

Vernadat, F. B. 2003. Enterprise modelling and integration, Springer.

Vernadat, F.B., 1996. Enterprise Modeling and Integration: Principles and Applications. Chapman & Hall London.

Vishal Naranje, S. Kumar, “A Low Cost Knowledge Base System Framework for Design of Deep Drawing Die”, World Academy of Science, Engineering and Technology, Vol:4, 2010, pp. 12-20.

Volkmann, J. W. & Westkämper, E. 2013. Cost Model for Digital Engineering Tools. Procedia CIRP, 7, 676-681.

Wahid, B., Ding, C., Ajaefobi, J., Agyapong-Kodua, K., Masood, T. & Weston, R. 2008. Enterprise modelling in support of methods based engineering: lean implementation in an SME.

Wang, L. (1992). "Analysis and design of fuzzy systems." Ph.D. Thesis University of Southern California. Signal and Image Processing Institute. USC-SIPI report 206.

Wasim A, Essam S, Hassan A, Ahmed A, Robert S, Rahman A. 2013. An innovative cost modelling system to support lean product and process development. International Journal of Advanced Manufacturing Technology, 65(1-4), pp.165–181.

Westkämper, E. 2007. Digital Manufacturing in the global Era. Digital Enterprise Technology. Springer.

Westkämper, E., Gottwald, B. & Fisser, F. 2005. Migration of the digital and virtual factory to reality. CIRP Journal of Manufacturing Systems, 34, 391-396.

Weston, R. H. 1999. A model-driven, component-based approach to reconfiguring manufacturing software systems. Int. J. of Operations and Production Management, Responsiveness in Manufacturing, 19, 834-855.

Westphal, P. & Sohal, A., 2016. Outsourcing decision-making: does the process matter? Production Planning & Control, 27(11), pp.894–908. Available at: http://www.tandfonline.com/doi/full/10.1080/09537287.2016.1159350 [Accessed December 8, 2016].

WIERDA, L. S. 1991. Linking Design, Process Planning and Cost Information by Feature-based Modelling. Journal of Engineering Design, 2, 3-19.

Wierda, L. S. 1991. Linking Design, Process Planning and Cost Information by Feature-based Modelling. Journal of Engineering Design, 2, 3-19.

Williams, T. J. & Li, H. 1997. The task force specification for GERAM and its fulfillment by PERA. Annual Reviews in Control, 21, 137-147.

Williams, T. J. 1994. The Purdue enterprise reference architecture. Computers in industry, 24, 141-158.

Williams, T. J. 2002. The Purdue Enterprise Reference Architecture. Instrument Society of America. Research Triangle Park, North Carolina, USA.

Williams, T., Bernus, P., Brosvic, J., Chen, D., Doumeingts, G., Nemes, L., Nevins, J., Vallespir, B., Vlietstra, J. & Zoetekouw, D. 1994. Architectures for integrating manufacturing activities and enterprises. Computers in Industry, 24, 111-139.

Williams, T.J., 1994. Computers i,n |ndustry. , 24, pp.141–158.

Wim J.C. Verhagen, Pablo Bermell-Garcia, Reinier E.C. van Dijk, Richard Curran, “A critical review of Knowledge-Based Engineering: An identification of research challenges”, vol. 26, 2012, pp. 5-15.

Wright, T. P. 1936. Factors affecting the cost of airplanes. Journal of the Aeronautical Sciences (Institute of the Aeronautical Sciences), 3.

Wu, C.-C. et al., 2009. Multiple attribute decision making assessment on the notebook computer industry performance. Journal of Discrete Mathematical Sciences and Cryptography Journal of Discrete Mathematical Sciences & Cryptography, 12(2), pp.217–237.

Y.-M. Deng, G. A. Britton and S. B. Tor, “A Design Perspective of Mechanical Function and its Object-Oriented Representation Scheme”, Engineering with Computers, 1998, vol. 14, pp. 309-320.

Yilmaz, S. & Seifert, C. M. 2011. Creativity through design heuristics: A case study of expert product design. Design Studies, 32, 384-415.

Ying-Han Wu, Heiu-Jou Shaw, “Document based knowledge base engineering method for ship basic design”, Ocean Engineering, vol. 38, 2011, pp. 1508–1521.

Yoon, J.-S., Shin, S.-J. & Suh, S.-H. 2011. A conceptual framework for the ubiquitous factory. International Journal of Production Research, 50, 2174-2189.

Zaeh, M. F., Moesl, J., Musiol, J. & Oefele, F. 2010. Material processing with remote technology revolution or evolution? Physics Procedia, 5, 19-33.

Zelm, M., Vernadat, F.B. & Kosanke, K., 1995. The Cimosa Business Modeling Process. Computers in Industry, 27(2), pp.123–142.

Zhu H, Menq C. B-rep model simplification by automatic fillet/round suppressing for efficient automatic feature recognition. Comput-Aided Des 2002;34(2):109–23.

Zoltowski, C. B., Oakes, W. C. & Cardella, M. E. 2012. Students' Ways of Experiencing Human-Centreed Design. Journal of Engineering Education, 101, 28-59.

APPENDIX A

Cost Modeller Cost Equations

A.1

```
/*
 * Name:    libAccounting_Assembly.csl
 * Author:  Ken Asare
 * Created: 02/06/2016
 * Purpose: Contains various functions for calculating process taxonomies
 * Status:  Complete
 * Note:    Formulas follow a standard naming convention. Any formula name ending in 0
 *          calls for the globally available accounting function. Any formula name
 *          ending in 1 or higher is process group specific.
 */


 /*
 * VARIABLE COSTS
 *
 * Function : GetLaborCost
 */
GetLaborCost_Assembly_ManualMIGWelding(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_Assembly_ManualSpotWelding(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_Assembly_MechanicalAssembly(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_Assembly_PickAndPlace(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_Assembly_RoboticMIGWelding(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_Assembly_RoboticSpotWelding(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_Assembly_WeldCleanUp(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_Assembly_WeldPrep(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_Assembly_PrepareDimples(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_Assembly_RemoteLaserWelding(laborTime, cycleTime, laborRate) = LaborCost0

     LaborCost0 = GetLaborCost(laborTime, cycleTime, laborRate)


 /*
 * Function : GetDirectOverheadCost
 */
GetDirectOverheadCost_Assembly_ManualMIGWelding(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate)
= DOHCost1
GetDirectOverheadCost_Assembly_ManualSpotWelding(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate)
= DOHCost1
GetDirectOverheadCost_Assembly_MechanicalAssembly(laborCost, cycleTime, laborTime, overheadMultiplier,
overheadRate) = DOHCost1
```

GetDirectOverheadCost_Assembly_PickAndPlace(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost1

GetDirectOverheadCost_Assembly_RoboticMIGWelding(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost1

GetDirectOverheadCost_Assembly_RoboticSpotWelding(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost1

GetDirectOverheadCost_Assembly_WeldCleanUp(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost1

GetDirectOverheadCost_Assembly_WeldPrep(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost1

GetDirectOverheadCost_Assembly_RemoteLaserWelding(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost1

*DOHCost1 = laborCost * overheadMultiplier + overheadRate * laborTime / SEC_PER_HR*

```
/*
 * Function : GetSetupCostPerPart
 */
```

GetSetupCostPerPart_Assembly_ManualMIGWelding(setupTimePerPart, laborRate, overheadRate) = SetupCost0
GetSetupCostPerPart_Assembly_ManualSpotWelding(setupTimePerPart, laborRate, overheadRate) = SetupCost0
GetSetupCostPerPart_Assembly_MechanicalAssembly(setupTimePerPart, laborRate, overheadRate) = SetupCost0
GetSetupCostPerPart_Assembly_PickandPlace(setupTimePerPart, laborRate, overheadRate) = SetupCost0
GetSetupCostPerPart_Assembly_RoboticMIGWelding(setupTimePerPart, laborRate, overheadRate) = SetupCost0
GetSetupCostPerPart_Assembly_RoboticSpotWelding(setupTimePerPart, laborRate, overheadRate) = SetupCost0
GetSetupCostPerPart_Assembly_WeldCleanUp(setupTimePerPart, laborRate, overheadRate) = SetupCost0
GetSetupCostPerPart_Assembly_WeldPrep(setupTimePerPart, laborRate, overheadRate) = SetupCost0
GetSetupCostPerPart_Assembly_RemoteLaserWelding(setupTimePerPart, laborRate, overheadRate) = SetupCost0

SetupCost0 = GetSetupCostPerPart(setupTimePerPart, laborRate, overheadRate)

```
/*
 * Function : GetExpendableToolingCostPerPart
 */
```

GetExpendableToolingCostPerPart_Assembly_ManualMIGWelding(weldWeight) = ETCost1
GetExpendableToolingCostPerPart_Assembly_ManualSpotWelding = ETCost0
GetExpendableToolingCostPerPart_Assembly_MechanicalAssembly = ETCost0
GetExpendableToolingCostPerPart_Assembly_PickAndPlace = ETCost0
GetExpendableToolingCostPerPart_Assembly_RoboticMIGWelding(weldWeight) = ETCost1
GetExpendableToolingCostPerPart_Assembly_RoboticSpotWelding = ETCost0
GetExpendableToolingCostPerPart_Assembly_WeldCleanUp = ETCost0
GetExpendableToolingCostPerPart_Assembly_WeldPrep = ETCost0
GetExpendableToolingCostPerPart_Assembly_RemoteLaserWelding = ETCost0

ETCost0 = GetExpendableToolingCostPerPart
*ETCost1 = weldWeight * machine.wireCost*

```
/*
 * Function : GetLogisticsCost
 */
GetLogisticsCost_Assembly_ManualMIGWelding = LogisticsCost0
GetLogisticsCost_Assembly_ManualSpotWelding = LogisticsCost0
GetLogisticsCost_Assembly_MechanicalAssembly = LogisticsCost0
GetLogisticsCost_Assembly_PickAndPlace = LogisticsCost0
GetLogisticsCost_Assembly_RoboticMIGWelding = LogisticsCost0
GetLogisticsCost_Assembly_RoboticSpotWelding = LogisticsCost0
GetLogisticsCost_Assembly_WeldCleanUp = LogisticsCost0
GetLogisticsCost_Assembly_WeldPrep = LogisticsCost0
GetLogisticsCost_Assembly_RemoteLaserWelding = LogisticsCost0
        LogisticsCost0 = GetLogisticsCost


/*
 * Function : GetAdditionalDirectCosts
 */
GetAdditionalDirectCosts_Assembly_ManualMIGWelding = ADCost0
GetAdditionalDirectCosts_Assembly_ManualSpotWelding = ADCost0
GetAdditionalDirectCosts_Assembly_MechanicalAssembly = ADCost0
GetAdditionalDirectCosts_Assembly_PickAndPlace = ADCost0
GetAdditionalDirectCosts_Assembly_RoboticMIGWelding = ADCost0
GetAdditionalDirectCosts_Assembly_RoboticSpotWelding = ADCost0
GetAdditionalDirectCosts_Assembly_WeldCleanUp = ADCost0
GetAdditionalDirectCosts_Assembly_WeldPrep = ADCost0
GetAdditionalDirectCosts_Assembly_RemoteLaserWelding = ADCost0

        ADCost0 = GetAdditionalDirectCosts


/*
 * Function : GetAdditionalAmortizedInvestment
 */
GetAdditionalAmortizedInvestment_Assembly_ManualMIGWelding = AAInvest0
GetAdditionalAmortizedInvestment_Assembly_ManualSpotWelding = AAInvest0
GetAdditionalAmortizedInvestment_Assembly_MechanicalAssembly = AAInvest0
GetAdditionalAmortizedInvestment_Assembly_PickAndPlace = AAInvest0
GetAdditionalAmortizedInvestment_Assembly_RoboticMIGWelding = AAInvest0
GetAdditionalAmortizedInvestment_Assembly_RoboticSpotWelding = AAInvest0
GetAdditionalAmortizedInvestment_Assembly_WeldCleanUp = AAInvest0
GetAdditionalAmortizedInvestment_Assembly_WeldPrep = AAInvest0
GetAdditionalAmortizedInvestment_Assembly_RemoteLaserWelding = AAInvest0

        AAInvest0 = GetAdditionalAmortizedInvestment
```

```
/*
 * Function : GetExtraCosts
 */
GetExtraCosts_Assembly_ManualMIGWelding = ECost0
GetExtraCosts_Assembly_ManualSpotWelding = ECost0
GetExtraCosts_Assembly_MechanicalAssembly = ECost0
GetExtraCosts_Assembly_PickAndPlace = ECost0
GetExtraCosts_Assembly_RoboticMIGWelding = ECost0
GetExtraCosts_Assembly_RoboticSpotWelding = ECost0
GetExtraCosts_Assembly_WeldCleanUp = ECost0
GetExtraCosts_Assembly_WeldPrep = ECost0
GetExtraCosts_Assembly_RemoteLaserWelding = ECost0

    ECost0 = GetExtraCosts


 /*
 * TIMES
 * Function : GetCycleTime
 */
GetCycleTime_Assembly_ManualMIGWelding(weldTime, laborTime, numOperators) = CycleTime2
GetCycleTime_Assembly_ManualSpotWelding(weldTime, laborTime, numOperators) = CycleTime2
GetCycleTime_Assembly_MechanicalAssembly(processTime, laborTime, numOperators) = CycleTime1
GetCycleTime_Assembly_PickAndPlace(processTime, laborTime, numOperators) = CycleTime1
GetCycleTime_Assembly_RoboticMIGWelding(weldTime, laborTime, numOperators)= CycleTime3
GetCycleTime_Assembly_RoboticSpotWelding(weldTime, laborTime, numOperators) = CycleTime3
GetCycleTime_Assembly_WeldCleanUp(processTime, laborTime, numOperators) = CycleTime4
GetCycleTime_Assembly_WeldPrep(processTime, laborTime, numOperators) = CycleTime4
GetCycleTime_Assembly_RemoteLaserWelding(processTime, laborTime, numOperators) = CycleTime5


    CycleTime1 = (laborTime / numOperators) * plant.cycleTimeAdjustmentFactor
    CycleTime2 = (weldTime / numOperators) * plant.cycleTimeAdjustmentFactor
    CycleTime3 = (weldTime / machine.numWelders) * plant.cycleTimeAdjustmentFactor
    CycleTime4 = (select sum(op.cycleTime) from childOps op) * plant.cycleTimeAdjustmentFactor
    CycleTime5 = weldTime * plant.cycleTimeAdjustmentFactor


/*
 * Function : GetLaborTime
 */
GetLaborTime_Assembly_ManualMIGWelding(processTime, numOperators, laborTimeStandard) = LaborTime1
GetLaborTime_Assembly_ManualSpotWelding(processTime, numOperators, laborTimeStandard) = LaborTime1
GetLaborTime_Assembly_MechanicalAssembly(processTime, numOperators, laborTimeStandard) = LaborTime2
GetLaborTime_Assembly_PickAndPlace(processTime, numOperators, unloadAsmFixtureTime, laborTimeStandard) =
LaborTime3
```

GetLaborTime_Assembly_RoboticMIGWelding(processTime, numOperators, laborTimeStandard) = LaborTime4
GetLaborTime_Assembly_RoboticSpotWelding(processTime, numOperators, laborTimeStandard) = LaborTime1
GetLaborTime_Assembly_WeldCleanUp(processTime, numOperators, laborTimeStandard) = LaborTime5
GetLaborTime_Assembly_WeldPrep(processTime, numOperators, laborTimeStandard) = LaborTime5
GetLaborTime_Assembly_RemoteLaserWelding(processTime, numOperators, laborTimeStandard) = LaborTime1

*LaborTime1 = weldTime * numOperators * laborTimeStandard*
*LaborTime2 = (select sum(op.cycleTime) from childOps op) * numOperators * laborTimeStandard*
*LaborTime3 = ((select sum(op.cycleTime) from childOps op) + unloadAsmFixtureTime) * numOperators * laborTimeStandard*
*LaborTime4 = {(((weldTime * numOperators) - pickAndPlaceTime) * laborTimeStandard) if (weldTime * numOperators > tackTime + pickAndPlaceTime)*
(tackTime) otherwise }
*LaborTime5 = processTime * numOperators * laborTimeStandard*

*/**
** Function : GetSetupTimePerPart*
**/*
GetSetupTimePerPart_Assembly_ManualMIGWelding(batchSize, cycleTime, batchSetupTime) = SetupTime0
GetSetupTimePerPart_Assembly_ManualSpotWelding(batchSize, cycleTime, batchSetupTime) = SetupTime0
GetSetupTimePerPart_Assembly_MechanicalAssembly(batchSize, cycleTime, batchSetupTime) = SetupTime0
GetSetupTimePerPart_Assembly_PickandPlace(batchSize, cycleTime, batchSetupTime) = SetupTime0
GetSetupTimePerPart_Assembly_RoboticMIGWelding(batchSize, cycleTime, batchSetupTime) = SetupTime0
GetSetupTimePerPart_Assembly_RoboticSpotWelding(batchSize, cycleTime, batchSetupTime) = SetupTime0
GetSetupTimePerPart_Assembly_WeldCleanUp(batchSize, cycleTime, batchSetupTime) = SetupTime0
GetSetupTimePerPart_Assembly_WeldPrep(batchSize, cycleTime, batchSetupTime) = SetupTime0
GetSetupTimePerPart_Assembly_RemoteLaserWelding(batchSize, cycleTime, batchSetupTime) = SetupTime0

SetupTime0 = GetSetupTimePerPart(batchSize, cycleTime, batchSetupTime)

*/**
** Function : GetLaborHandlingTime*
**/*
GetLaborHandlingTime_Assembly_ManualMIGWelding = LHTime0
GetLaborHandlingTime_Assembly_ManualSpotWelding = LHTime0
GetLaborHandlingTime_Assembly_MechanicalAssembly = LHTime0
GetLaborHandlingTime_Assembly_PickAndPlace = LHTime0
GetLaborHandlingTime_Assembly_RoboticMIGWelding = LHTime0
GetLaborHandlingTime_Assembly_RoboticSpotWelding = LHTime0
GetLaborHandlingTime_Assembly_WeldCleanUp = LHTime0
GetLaborHandlingTime_Assembly_WeldPrep = LHTime0
GetLaborHandlingTime_Assembly_RemoteLaserWelding = LHTime0

LHTime0 = GetLaborHandlingTime

```
/*
 * PERIOD COSTS
 *
 * Function : GetPeriodOverhead
 */
GetPeriodOverhead_Assembly_ManualMIGWelding(periodOverheadCoefficient) = POH0
GetPeriodOverhead_Assembly_ManualSpotWelding(periodOverheadCoefficient) = POH0
GetPeriodOverhead_Assembly_MechanicalAssembly(periodOverheadCoefficient) = POH0
GetPeriodOverhead_Assembly_PickAndPlace(periodOverheadCoefficient) = POH0
GetPeriodOverhead_Assembly_RoboticMIGWelding(periodOverheadCoefficient) = POH0
GetPeriodOverhead_Assembly_RoboticSpotWelding(periodOverheadCoefficient) = POH0
GetPeriodOverhead_Assembly_WeldCleanUp(periodOverheadCoefficient) = POH0
GetPeriodOverhead_Assembly_WeldPrep(periodOverheadCoefficient) = POH0
GetPeriodOverhead_Assembly_RemoteLaserWelding(periodOverheadCoefficient) = POH0

    POH0 = GetPeriodOverhead(periodOverheadCoefficient)
```

A.2

```
//*******************************************************
// Name: libBatchSetup.csl
// Date: 05/07/2016
// Purpose: Compute setup cost per part
//*********************************************************
// Calculate setup cost per part by allocating a labor and overhead cost to the setup time per part
setupCostPerPartTotal = setupLaborCostPerPart + setupOverheadCostPerPart
setupLaborCostPerPart = setupTimePerPart * laborRate
setupOverheadCostPerPart =  setupTimePerPart *  overheadRate
setupTimePerPart = {       (batchSetupTime / batchSize) if (cycleTime > 0)
                                                                 (0) otherwise }
```

A.3

```
/*
 * Name:    libMaterialCost_Assembly.csl
 * Author:  Ken Asare
 * Created: 12/07/2016
 * Purpose: Contains various functions for calculating material cost
 * Status:  Complete
 * Note:    Formulas follow a standard naming convention. Any formula name ending in 0
 *          calls for the globally available accounting function. Any formula name
 *          ending in 1 or higher is process group specific.
 */

/*
 * VARIABLE COSTS
 * materialCost: material cost per part, $/part
 */


/*
 * Function : GetMaterialCost
 * Returns  : Part material cost, $/part
 */
GetMaterialCost_Assembly_ManualMIGWelding = MCost0
GetMaterialCost_Assembly_ManualSpotWelding = MCost0
GetMaterialCost_Assembly_MechanicalAssembly = MCost0
```

GetMaterialCost_Assembly_PickandPlace = MCost0

GetMaterialCost_Assembly_RoboticMIGWelding = MCost0

GetMaterialCost_Assembly_RoboticSpotWelding = MCost0

GetMaterialCost_Assembly_WeldCleanUp = MCost0

GetMaterialCost_Assembly_WeldPrep = MCost0

GetMaterialCost_Assembly_RemoteLaserWelding = MCost0

MCost0 = UNUSED

A.4

```
/*
 * Name:    libCommonAccounting.csl
 * Author:  Ken Asare
 * Created: 02/07/2016
 * Purpose: Contains various functions for calculating process taxonomies
 * Status:  In progress
 */

/*
 * VARIABLE COSTS
 * laborCost: labor cost per part, $/part
 * directOverheadCost: overhead cost per part (from the machine $/hr), $/part
 * expendableToolingCostPerPart: cost of inserts and other consumable materials per part,
$/part
 * logisticsCost: logistics cost per part, $/part
 * additionalDirectCosts: additional direct costs per part, $/part
 * additionalAmortizedInvestment: additional amortized investment per part, $/part
 * extraCosts: extra costs per part, $/part
 *
 * TIMES
 * cycleTime: base process cycle time, s
 * laborTime: base labor time per cycle, s
 *
 * PERIOD COSTS
 * periodOverhead: period overhead cost per part, $/part
 */
/*
 * FUNCTIONS FOR INTERFACE VARIABLES
 *
 * Function : GetMachineNumOperators
```

```
 * Returns  : Operators or fraction of an operator assigned to a particular machine
 * Note     : Checks for missing data in machine table and fails with with the name of the offending machine.
 */
GetMachineNumOperators = { setup.numOperators if (setup.numOperators != null)
                    fail(msg(FLT, 'machine_numOperators:Bad machine data. numOperators is null. machine=', machine_name, FRT)) if (machine.numOperators == null)
                    fail(msg(FLT, 'machine_numOperators:Bad machine data. numOperators must be >= 0. machine=', machine_name, FRT)) if (machine.numOperators < 0)
                    machine.numOperators otherwise }
/*
 * Function : GetMachineLaborRate
 * Returns  : Labor rate for one full person, $/hr
 * Note     : Checks for missing data in machine table and fails with with the name of the offending machine.
 */
GetMachineLaborRate = { fail(msg(FLT, 'machine_laborRate:Bad machine data. laborRate is null. machine=', machine_name, FRT)) if (machine.workCenterLaborRate == null)
                  fail(msg(FLT, 'machine_laborRate:Bad machine data. laborRate must be >= 0. machine=', machine_name, FRT)) if (machine.workCenterLaborRate < 0)
                  machine.workCenterLaborRate * plant.laborRateAdjustmentFactor otherwise }
/*
 * Function : GetMachineOverheadRate
 * Returns  : Direct Overhead rate for the machine, $/hr
 * Note     : Checks for missing data in machine table and fails with with the name of the offending machine.
 */
GetMachineOverheadRate = { fail(msg(FLT, 'machine_overheadRate:Bad machine data. overheadRate is null. machine=', machine_name, FRT)) if (machine.workCenterOverheadRate == null)
                    fail(msg(FLT, 'machine_overheadRate:Bad machine data. overheadRate must be >= 0. machine=', machine_name, FRT)) if (machine.workCenterOverheadRate < 0)
                    machine.workCenterOverheadRate * plant.overheadRateAdjustmentFactor otherwise }
/*
 * Function : GetMachineOverheadMultiplier
 * Returns  : Labor cost multiplier used by some factory accounting systems. When non-zero, it adds an additional cost term to overhead.
```

```
 * Note    : Checks for missing data in machine table and fails with with the name of the offending machine.
 */
GetMachineOverheadMultiplier = { fail(msg(FLT, 'machine_workCenterOverheadMultiplier:Bad machine data. overheadMultiplier is null. machine=', machine_name, FRT)) if (machine.workCenterOverheadMultiplier == null)
                    fail(msg(FLT, 'machine_workCenterOverheadMultiplier:Bad machine data. overheadMultiplier must be >= 0. machine=', machine_name, FRT)) if (machine.workCenterOverheadMultiplier < 0)
                    machine.workCenterOverheadMultiplier otherwise }
/*
 * Function : GetPeriodOverheadCoefficient
 * Returns  : Period Overhead Coefficient
 * Note    : Checks for missing data in machine table and fails with with the name of the offending machine.
 */
GetPeriodOverheadCoefficient = { fail(msg(FLT, 'machine_periodOverheadCoefficient:Bad machine data. Period overhead coefficient is null. machine=', machine_name, FRT)) if (machine.periodOverheadCoefficient == null)
                    fail(msg(FLT, 'machine_periodOverheadCoefficient:Bad machine data. Period overhead coefficient must be >= 0. machine=', machine_name, FRT)) if (machine.periodOverheadCoefficient < 0)
                    machine.periodOverheadCoefficient otherwise }
/*
 * Function : GetMachineLaborTimeStandard
 * Purpose  :
 * Returns  : Time standard for the machine, 0 to 1
 * Note    : Checks for missing data in machine table and fails with with the name of the offending machine.
 */
GetMachineLaborTimeStandard = { fail(msg(FLT, 'machine_laborTimeStandard:Bad machine data. Labor time standard is null. machine=', machine_name, FRT)) if ( machine.laborTimeStandard == null)
                    fail(msg(FLT, 'machine_laborTimeStandard:Bad machine data. Labor time standard must be >= 0. machine=', machine_name, FRT)) if  (machine.laborTimeStandard < 0)
                    machine.laborTimeStandard otherwise }
/*
```

```
 * Function : GetBatchSetupTime
 * Returns  : Time to set up the machine for a batch of parts, hr
 * Note    : Checks for missing data in machine table and fails with with the name of the offending
machine.
 */
GetBatchSetupTime = { setup.batchSetupTime if (setup.batchSetupTime != null) and
(setup.batchSetupTime >= 0)
                fail(msg(FLT, 'machine_setupTime:Bad machine data. batchSetupTime is null.
machine=', machine_name, FRT)) if (machine.setupTime == null)
                fail(msg(FLT, 'machine_setupTime:Bad machine data. batchSetupTime must be
>= 0. machine=', machine_name, FRT)) if (machine.setupTime < 0)
                machine.setupTime otherwise }
/*
 * VARIABLE COSTS
 *
 * Function : GetLaborCost
 * Accepts  : Paid time for direct labor needed to manufacture the part or assembly, s
 * Returns  : Cost of direct labor needed to manufacture the part or assembly, $/part
 */
GetLaborCost(laborTime, cycleTime, laborRate) = laborRate * laborTime / SEC_PER_HR
/*
 * Function : GetDirectOverheadCost
 * Accepts  : Cost of direct labor needed to manufacture the part or assembly, $/part
 *          Total time for one complete cycle of the applicable machine to make a single part, s
 * Returns  : Direct overhead cost for one part, $/part
 */
GetDirectOverheadCost(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate)
= (laborCost * overheadMultiplier) + (overheadRate * cycleTime / SEC_PER_HR)
/*
 * Function : GetSetupCostPerPart
 * Accepts  : Average setup time per part, hr
 * Returns  : Batch setup cost per part, $/part
 */
GetSetupCostPerPart(setupTimePerPart, laborRate, overheadRate) = (setupTimePerPart *
laborRate) + (setupTimePerPart * overheadRate)
/*
 * Function : GetExpendableToolingCostPerPart
 * Accepts  : childOps - set of all operations associated with the process
```

```
 * Returns  : Cost of non-custom tooling that is consumed during manufacturing of a part, $/part
 */
GetExpendableToolingCostPerPart = UNUSED
/*
 * Function : GetLogisticsCost
 * Accepts  : childOps - set of all operations associated with the process
 * Returns  : Cost of transportation and tariffs between facilities, $/part
 */
GetLogisticsCost = UNUSED
/*
 * Function : GetAdditionalDirectCosts
 * Accepts  : childOps - set of all operations associated with the process
 * Returns  : Other costs that can be specifically associated with the manufacture of a given design, $/part
 */
GetAdditionalDirectCosts = UNUSED
/*
 * Function : GetAdditionalAmortizedInvestment
 * Accepts  : childOps - set of all operations associated with the process
 * Returns  : Additional amortized sum of all fixed costs not otherwise accounted for, $/part
 */
GetAdditionalAmortizedInvestment =  UNUSED
/*
 * Function : GetExtraCosts
 * Accepts  : childOps - set of all operations associated with the process
 * Returns  : Total of extra miscellaneous costs, $/part
 */
GetExtraCosts =  UNUSED
 /*
 * TIMES
 *
 * Function : GetCycleTime
 * Accepts  : Total process cycle time, s
 * Returns  : Time for one complete cycle of the applicable machines to make a single part, s
 */
GetCycleTime(processTime, numOperators) = processTime * plant.cycleTimeAdjustmentFactor
/*
```

```
 * Function : GetLaborTime
 * Accepts  : Time for one complete cycle of the applicable machines to make a single part, s
 * Returns  : Paid time for direct labor needed to manufacture the part or assembly, s
 */
GetLaborTime(processTime, cycleTime, numOperators, laborTimeStandard) = cycleTime * numOperators * laborTimeStandard
/*
 * Function : GetSetupTimePerPart
 * Accepts  : batchSize - average size of a production batch
 *           cycleTime - sum of all process/operations times, s
 * Returns  : Setup time per part, hr/part
 */
GetSetupTimePerPart(batchSize, cycleTime, batchSetupTime) = { (0) if (cycleTime == 0)
                                        (batchSetupTime / batchSize) otherwise }
/*
 * Function : GetLaborHandlingTime
 * Accepts  : childOps - set of all operations associated with the process
 * Returns  : Labor handling time, s
 */
GetLaborHandlingTime = UNUSED
 /*
 * PERIOD COSTS
 *
 * Function : GetPeriodOverhead
 * Accepts  : Period overhead coefficient
 * Returns  : Indirect overhead that is allocated to the part cost, $/part
 */
GetPeriodOverhead(periodOverheadCoefficient) = periodOverheadCoefficient * part.productionLife
```

A.5

```
rule1 = {  true if   (isExtruding(op1) and isCommonBending(op2))
  and (relation.type == RelationType.LIES_OUTSIDE and liesOutside(relation, op1, op2)) and not isOrthogonal(op1) or _
                                  (isGenericCoining(op1) and isCommonBending(op2)) and (relation.type == RelationType.LIES_OUTSIDE and liesOutside(relation, op1, op2)) and not isOrthogonal(op1) or _
                                  (isPiercing(op1)
  and isCommonBending(op2))     and (relation.type == RelationType.LIES_OUTSIDE and
```

liesOutside(relation, op1, op2)) and not isOrthogonal(op1) or _

(isTapping(op1) and isCommonBending(op2)) and (relation.type == RelationType.LIES_OUTSIDE and liesOutside(relation, op1, op2)) and not isOrthogonal(op1) or _

(isEmbossing(op1) and isGenericBending(op2) and (relation.type == RelationType.LIES_OUTSIDE and liesOutside(relation, op1, op2))) or _

(isBending(op1) and isBending(op2) and (relation.type == RelationType.LIES_OUTSIDE and liesOutside(relation, op1, op2))) or _

(isBending(op1) and isCamPiercing(op2) and (relation.type == RelationType.LIES_OUTSIDE and liesOutside(relation, op2, op1))) or _

(isCamActionBending(op1) and isBending(op2) and (relation.type == RelationType.LIES_OUTSIDE and liesOutside(relation, op1, op2))) or _

(isCamActionBending(op1) and isRestriking(op2) and (relation.type == RelationType.LIES_OUTSIDE and liesOutside(relation, op1, op2))) or _

(isCutoff(op2)) or _

(isPiloting(op1)) or _

((isEmbossing(op1) or isRestriking(op1) or isDrawing(op1) or isDeepDrawing(op1)) and isRePiloting(op2)) or _

(isEmbossing(op1) and isRestriking(op2)) or _

(isEdgeTrimming(op1) and isCamPiercing(op2)) or _

(isEdgeTrimming(op1) and isEmbossing(op2)) or _

(isEdgeTrimming(op1) and isBending(op2)) or _

(isEdgeTrimming(op1) and (isDrawing(op2) or isDeepDrawing(op2))) or _

(isFullBlanking(op1) and isEmbossing(op2)) or _

(isFullBlanking(op1) and isBending(op2)) or _

(isFullBlanking(op1) and

(isDrawing(op2) or isDeepDrawing(op2))) or _

(isFullBlanking(op1) and (isGenericForming(op2) or isSideActionForming(op2))) or _

(isFullBlanking(op1) and isCamPiercing(op2)) or _

(isFullBlanking(op1) and isExtruding(op2)) or _

(isTrimming(op1) and (isGenericForming(op2) or isSideActionForming(op2))) or _

(isTrimming(op1) and isCamPiercing(op2)) or _

(isTrimming(op1) and isExtruding(op2)) or _

(isTrimming(op1) and (isDrawing(op2) or isDeepDrawing(op2))) or _

(isSideActionForming(op1) and isCamPiercing(op2)) or _

(isCamActionBending(op1) and isCamPiercing(op2)) or _

(isDownOverBending(op1) and isCamPiercing(op2)) or _

(isCountersinking(op1) and isCamPiercing(op2)) or _

(isRestriking(op1) and isCamPiercing(op2)) or _

(isBasicForming(op1) and isCamActionBending(op2) and (relation.type == RelationType.LIES_OUTSIDE and not liesOutside(relation, op2, op1))) or _

(isBasicForming(op1) and (isCamPiercing(op2) or isCamActionTrimming(op2))) or _

(isPiercing(op1) and isBending(op2) and (not holeIsAccessible(op1.artifact))) or _

(isCamPiercing(op1) and isPiercing(op2) and (pierceHolesAfterForming and isOrthogonal(op2))) or _

(isAnyFormingOperation(op1) and isPiercing(op2) and (pierceHolesAfterForming and isOrthogonal(op2))) or _

(isOrthogonalHoleMaking(op1) and isAnyFormingOperation(op2) and pierceHolesBeforeForming) or _

(isPostFormTrimming(op1) and (isPiercing(op2) or isCamPiercing(op2))) or _

(isCamActionTrimming(op1) and (isPiercing(op2) or isCamPiercing(op2))) or _

((isUniqueForming(op1) or isGeneralBending(op1)) and isCamActionTrimming(op2)) or _

((isEmbossing(op1) or isRestriking(op1)) and isTapping(op2)) or _

(isTapping(op1) and isCamPiercing(op2)) or _

((isEmbossing(op1) or isRestriking(op1) or isDrawing(op1) or isDeepDrawing(op1) or isBending(op1) or isCamActionBending(op1)) and isPostFormTrimming(op2))

false otherwise }

isFlangedHole(op) = { true if (op.artifact.isFlanged == true) false otherwise }

isCountersunkHole(op) = { true if (op.artifactTypeName == 'SimpleHole' and op.artifact.isCountersunk == true) false otherwise }

isOpSpoiled(op) = (not isGenericEmbossing(op) and (precedes(op, 'Embossing') or precedes(op, 'ShearEmbossing')))

isOrthogonal(op) = isHoleOrthogonalToMainSurf(op.artifact)

A.6

*stockLengthCalc = { (blank.serLength + (2 * edgeMargin) + pitchMargin) if (blankIsWiderThanTall and isLengthWiseOrientation and isCenterCarrierStrip)*

*(blank.serLength + (2 * edgeMargin)) if (blankIsWiderThanTall and isLengthWiseOrientation and not isCenterCarrierStrip)*

*(blank.serWidth + (2 * edgeMargin) + pitchMargin) if (blankIsWiderThanTall and isWidthWiseOrientation and isCenterCarrierStrip)*

*(blank.serWidth + (2 * edgeMargin)) if (blankIsWiderThanTall and isWidthWiseOrientation and not isCenterCarrierStrip)*

*(blank.serWidth + (2 * edgeMargin) +*

pitchMargin) if (blankIsTallerThanWide and isLengthWiseOrientation and isCenterCarrierStrip)

*(blank.serWidth + (2 * edgeMargin)) if (blankIsTallerThanWide and isLengthWiseOrientation and not isCenterCarrierStrip)*

*(blank.serLength + (2 * edgeMargin) + pitchMargin) if (blankIsTallerThanWide and isWidthWiseOrientation and isCenterCarrierStrip)*

*(blank.serLength + (2 * edgeMargin)) if (blankIsTallerThanWide and isWidthWiseOrientation and not isCenterCarrierStrip)*

0 otherwise }

*stockWidthCalc = { ((numConcurrentParts * blank.serWidth) + totalWidthMargin) if (blankIsWiderThanTall and isLengthWiseOrientation and isCenterCarrierStrip)*

*((numConcurrentParts * blank.serWidth) + totalWidthMargin) if (blankIsWiderThanTall and isLengthWiseOrientation and isDoubleEdgeCarrierStrip)*

*((numConcurrentParts * blank.serWidth) + totalWidthMargin) if (blankIsWiderThanTall and isLengthWiseOrientation and isDoubleEdgeCarrierStrip)*

*((numConcurrentParts * blank.serLength) + totalWidthMargin) if (blankIsWiderThanTall and isWidthWiseOrientation and isCenterCarrierStrip)*

*((numConcurrentParts * blank.serLength) + totalWidthMargin) if (blankIsWiderThanTall and isWidthWiseOrientation and isDoubleEdgeCarrierStrip)*

*((numConcurrentParts * blank.serLength) + totalWidthMargin) if (blankIsWiderThanTall and isWidthWiseOrientation and isDoubleEdgeCarrierStrip)*

*((numConcurrentParts * blank.serLength) + totalWidthMargin) if (blankIsTallerThanWide and isLengthWiseOrientation and isCenterCarrierStrip)*

*((numConcurrentParts * blank.serLength) + totalWidthMargin) if (blankIsTallerThanWide and isLengthWiseOrientation and isSingleEdgeCarrierStrip)*

*((numConcurrentParts * blank.serLength) + totalWidthMargin) if (blankIsTallerThanWide and isLengthWiseOrientation and isDoubleEdgeCarrierStrip)*

*((numConcurrentParts **

blank.serWidth) + totalWidthMargin) if (blankIsTallerThanWide and isWidthWiseOrientation and isCenterCarrierStrip)

*((numConcurrentParts * blank.serWidth) + totalWidthMargin) if (blankIsTallerThanWide and isWidthWiseOrientation and isSingleEdgeCarrierStrip)*

*((numConcurrentParts * blank.serWidth) + totalWidthMargin) if (blankIsTallerThanWide and isWidthWiseOrientation and isDoubleEdgeCarrierStrip)*

0 otherwise }

isWidthWiseOrientation = { true if setup.xyOrientation == 'widthWise' false otherwise }

isLengthWiseOrientation = { true if setup.xyOrientation == 'lengthWise' false otherwise }

defaultXYOrientation = { 'lengthWise' if blankIsWiderThanTall

'widthWise' otherwise }

blankIsWiderThanTall = { true if blank.serLength > blank.serWidth false otherwise }

blankIsTallerThanWide = { true if blank.serWidth > blank.serLength false otherwise }

A.7 Stock Size Calculation

*stockLengthCalc = { (blank.serLength + (2 * edgeMargin) + pitchMargin) if (blankIsWiderThanTall and isLengthWiseOrientation and isCenterCarrierStrip)*

*(blank.serLength + (2 * edgeMargin)) if (blankIsWiderThanTall and isLengthWiseOrientation and not isCenterCarrierStrip)*

*(blank.serWidth + (2 * edgeMargin) + pitchMargin) if (blankIsWiderThanTall and isWidthWiseOrientation and isCenterCarrierStrip)*

*(blank.serWidth + (2 * edgeMargin)) if (blankIsWiderThanTall and isWidthWiseOrientation and not isCenterCarrierStrip)*

*(blank.serWidth + (2 * edgeMargin) + pitchMargin) if (blankIsTallerThanWide and isLengthWiseOrientation and isCenterCarrierStrip)*

*(blank.serWidth + (2 * edgeMargin)) if (blankIsTallerThanWide and isLengthWiseOrientation and not isCenterCarrierStrip)*

*(blank.serLength + (2 * edgeMargin) + pitchMargin) if (blankIsTallerThanWide and isWidthWiseOrientation and*

isCenterCarrierStrip)

*(blank.serLength + (2 * edgeMargin)) if (blankIsTallerThanWide and isWidthWiseOrientation and not isCenterCarrierStrip)*

0 otherwise }

*stockWidthCalc = { ((numConcurrentParts * blank.serWidth) + totalWidthMargin) if (blankIsWiderThanTall and isLengthWiseOrientation and isCenterCarrierStrip)*

*((numConcurrentParts * blank.serWidth) + totalWidthMargin) if (blankIsWiderThanTall and isLengthWiseOrientation and isDoubleEdgeCarrierStrip)*

*((numConcurrentParts * blank.serWidth) + totalWidthMargin) if (blankIsWiderThanTall and isLengthWiseOrientation and isDoubleEdgeCarrierStrip)*

*((numConcurrentParts * blank.serLength) + totalWidthMargin) if (blankIsWiderThanTall and isWidthWiseOrientation and isCenterCarrierStrip)*

*((numConcurrentParts * blank.serLength) + totalWidthMargin) if (blankIsWiderThanTall and isWidthWiseOrientation and isDoubleEdgeCarrierStrip)*

*((numConcurrentParts * blank.serLength) + totalWidthMargin) if (blankIsWiderThanTall and isWidthWiseOrientation and isDoubleEdgeCarrierStrip)*

*((numConcurrentParts * blank.serLength) + totalWidthMargin) if (blankIsTallerThanWide and isLengthWiseOrientation and isCenterCarrierStrip)*

*((numConcurrentParts * blank.serLength) + totalWidthMargin) if (blankIsTallerThanWide and isLengthWiseOrientation and isSingleEdgeCarrierStrip)*

*((numConcurrentParts * blank.serLength) + totalWidthMargin) if (blankIsTallerThanWide and isLengthWiseOrientation and isDoubleEdgeCarrierStrip)*

*((numConcurrentParts * blank.serWidth) + totalWidthMargin) if (blankIsTallerThanWide and isWidthWiseOrientation and isCenterCarrierStrip)*

*((numConcurrentParts * blank.serWidth) + totalWidthMargin) if (blankIsTallerThanWide and isWidthWiseOrientation and isSingleEdgeCarrierStrip)*

*((numConcurrentParts **

blank.serWidth) + totalWidthMargin) if (blankIsTallerThanWide and isWidthWiseOrientation and isDoubleEdgeCarrierStrip)

0 otherwise }

isWidthWiseOrientation = { true if setup.xyOrientation == 'widthWise' false otherwise }
isLengthWiseOrientation = { true if setup.xyOrientation == 'lengthWise' false otherwise }
defaultXYOrientation = { 'lengthWise' if blankIsWiderThanTall
'widthWise' otherwise }

blankIsWiderThanTall = { true if blank.serLength > blank.serWidth false otherwise }
blankIsTallerThanWide = { true if blank.serWidth > blank.serLength false otherwise }

A.8 Sheet Metal CSL

```
/*
 * Name:    libAccounting_SheetMetal.csl
 * Author:  Ken Asare
 * Created: 02/02/2016
 * Purpose: Contains various functions for calculating process taxonomies
 * Status:  Complete
 * Note:    Formulas follow a standard naming convention. Any formula name ending in 0
 *          calls for the globally available accounting function. Any formula name
 *          ending in 1 or higher is process group specific.
 */


/*
 * VARIABLE COSTS
 *
 * Function : GetLaborCost
 */
GetLaborCost_SheetMetal_3DLaser(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_SheetMetal_BendBrake(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_SheetMetal_CTL(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_SheetMetal_Deburr(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_SheetMetal_Deslag(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_SheetMetal_GenericPress(laborTime, cycleTime, laborRate) = LaborCost0
```

GetLaborCost_SheetMetal_LaserCut(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_SheetMetal_MaterialStock(laborTime, cycleTime, laborRate) = LaborCost1
GetLaborCost_SheetMetal_OxyFuelCut(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_SheetMetal_PlasmaCut(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_SheetMetal_ProgressiveDie(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_SheetMetal_Shear(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_SheetMetal_StdPress(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_SheetMetal_TandemPress(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_SheetMetal_TransferPress(laborTime, cycleTime, laborRate) = LaborCost0
GetLaborCost_SheetMetal_TurretPress(laborTime, cycleTime, laborRate) = LaborCost0

LaborCost0 = GetLaborCost(laborTime, cycleTime, laborRate)
LaborCost1 = UNUSED

*/**
** Function : GetDirectOverheadCost*
**/*

GetDirectOverheadCost_SheetMetal_3DLaser(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost0
GetDirectOverheadCost_SheetMetal_BendBrake(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost0
GetDirectOverheadCost_SheetMetal_CTL(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost0
GetDirectOverheadCost_SheetMetal_Deburr(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost0
GetDirectOverheadCost_SheetMetal_Deslag(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost0
GetDirectOverheadCost_SheetMetal_GenericPress(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost0
GetDirectOverheadCost_SheetMetal_LaserCut(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost0
GetDirectOverheadCost_SheetMetal_MaterialStock(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost1
GetDirectOverheadCost_SheetMetal_OxyFuelCut(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost0
GetDirectOverheadCost_SheetMetal_PlasmaCut(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost0

GetDirectOverheadCost_SheetMetal_ProgressiveDie(resultDirectOverheadCost, stationCount, overheadMultiplier, overheadRate) = DOHCost2
GetDirectOverheadCost_SheetMetal_Shear(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost0
GetDirectOverheadCost_SheetMetal_StdPress(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost0
GetDirectOverheadCost_SheetMetal_TandemPress(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost0
GetDirectOverheadCost_SheetMetal_TransferPress(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost0
GetDirectOverheadCost_SheetMetal_TurretPress(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate) = DOHCost0

DOHCost0 = GetDirectOverheadCost(laborCost, cycleTime, laborTime, overheadMultiplier, overheadRate)
DOHCost1 = UNUSED
DOHCost2 = resultDirectOverheadCost / stationCount

*/**
** Function : GetSetupCostPerPart*
**/*
GetSetupCostPerPart_SheetMetal_3DLaser(setupTimePerPart, laborRate, overheadRate) = SetupCost0
GetSetupCostPerPart_SheetMetal_BendBrake(setupTimePerPart, laborRate, overheadRate) = SetupCost0
GetSetupCostPerPart_SheetMetal_CTL(setupTimePerPart, laborRate, overheadRate) = SetupCost0
GetSetupCostPerPart_SheetMetal_Deburr(setupTimePerPart, laborRate, overheadRate) = SetupCost0
GetSetupCostPerPart_SheetMetal_Deslag(setupTimePerPart, laborRate, overheadRate) = SetupCost0
GetSetupCostPerPart_SheetMetal_GenericPress(setupTimePerPart, laborRate, overheadRate) = SetupCost0
GetSetupCostPerPart_SheetMetal_LaserCut(setupTimePerPart, laborRate, overheadRate) = SetupCost0
GetSetupCostPerPart_SheetMetal_MaterialStock(setupTimePerPart, laborRate, overheadRate) = SetupCost0
GetSetupCostPerPart_SheetMetal_OxyFuelCut(setupTimePerPart, laborRate, overheadRate)

= SetupCost0

GetSetupCostPerPart_SheetMetal_PlasmaCut(setupTimePerPart, laborRate, overheadRate) = SetupCost0

GetSetupCostPerPart_SheetMetal_ProgressiveDie(setupTimePerPart, laborRate, overheadRate) = SetupCost0

GetSetupCostPerPart_SheetMetal_Shear(setupTimePerPart, laborRate, overheadRate) = SetupCost0

GetSetupCostPerPart_SheetMetal_StdPress(setupTimePerPart, laborRate, overheadRate) = SetupCost0

GetSetupCostPerPart_SheetMetal_TandemPress(setupTimePerPart, laborRate, overheadRate) = SetupCost0

GetSetupCostPerPart_SheetMetal_TransferPress(setupTimePerPart, laborRate, overheadRate) = SetupCost0

GetSetupCostPerPart_SheetMetal_TurretPress(setupTimePerPart, laborRate, overheadRate) = SetupCost0

SetupCost0 = GetSetupCostPerPart(setupTimePerPart, laborRate, overheadRate)

```
/*
 * Function : GetExpendableToolingCostPerPart
 */
GetExpendableToolingCostPerPart_SheetMetal_3DLaser = ETCost0
GetExpendableToolingCostPerPart_SheetMetal_BendBrake = ETCost0
GetExpendableToolingCostPerPart_SheetMetal_CTL = ETCost0
GetExpendableToolingCostPerPart_SheetMetal_Deburr = ETCost0
GetExpendableToolingCostPerPart_SheetMetal_Deslag = ETCost0
GetExpendableToolingCostPerPart_SheetMetal_GenericPress = ETCost0
GetExpendableToolingCostPerPart_SheetMetal_LaserCut = ETCost0
GetExpendableToolingCostPerPart_SheetMetal_MaterialStock = ETCost0
GetExpendableToolingCostPerPart_SheetMetal_OxyFuelCut = ETCost0
GetExpendableToolingCostPerPart_SheetMetal_PlasmaCut = ETCost0
GetExpendableToolingCostPerPart_SheetMetal_ProgressiveDie = ETCost1
GetExpendableToolingCostPerPart_SheetMetal_Shear = ETCost0
GetExpendableToolingCostPerPart_SheetMetal_StdPress = ETCost0
GetExpendableToolingCostPerPart_SheetMetal_TandemPress = ETCost0
GetExpendableToolingCostPerPart_SheetMetal_TransferPress = ETCost0
GetExpendableToolingCostPerPart_SheetMetal_TurretPress(expendableToolingCost,
```

totalProductionVolume) = ETCost2

ETCost0 = GetExpendableToolingCostPerPart

ETCost1 = results.expendableToolingCostPerPart

ETCost2 = expendableToolingCost / totalProductionVolume

```
/*
 * Function : GetLogisticsCost
 */
GetLogisticsCost_SheetMetal_3DLaser = LogisticsCost0
GetLogisticsCost_SheetMetal_BendBrake = LogisticsCost0
GetLogisticsCost_SheetMetal_CTL = LogisticsCost0
GetLogisticsCost_SheetMetal_Deburr = LogisticsCost0
GetLogisticsCost_SheetMetal_Deslag = LogisticsCost0
GetLogisticsCost_SheetMetal_GenericPress = LogisticsCost0
GetLogisticsCost_SheetMetal_LaserCut = LogisticsCost0
GetLogisticsCost_SheetMetal_MaterialStock = LogisticsCost0
GetLogisticsCost_SheetMetal_OxyFuelCut = LogisticsCost0
GetLogisticsCost_SheetMetal_PlasmaCut = LogisticsCost0
GetLogisticsCost_SheetMetal_ProgressiveDie = LogisticsCost1
GetLogisticsCost_SheetMetal_Shear = LogisticsCost0
GetLogisticsCost_SheetMetal_StdPress = LogisticsCost0
GetLogisticsCost_SheetMetal_TandemPress = LogisticsCost0
GetLogisticsCost_SheetMetal_TransferPress = LogisticsCost0
GetLogisticsCost_SheetMetal_TurretPress = LogisticsCost0

    LogisticsCost0 = GetLogisticsCost
    LogisticsCost1 = results.logisticsCost


/*
 * Function : GetAdditionalDirectCosts
 */
GetAdditionalDirectCosts_SheetMetal_3DLaser = ADCost0
GetAdditionalDirectCosts_SheetMetal_BendBrake = ADCost0
GetAdditionalDirectCosts_SheetMetal_CTL = ADCost0
GetAdditionalDirectCosts_SheetMetal_Deburr = ADCost0
```

```
GetAdditionalDirectCosts_SheetMetal_Deslag = ADCost0
GetAdditionalDirectCosts_SheetMetal_GenericPress = ADCost0
GetAdditionalDirectCosts_SheetMetal_LaserCut = ADCost0
GetAdditionalDirectCosts_SheetMetal_MaterialStock = ADCost0
GetAdditionalDirectCosts_SheetMetal_OxyFuelCut = ADCost0
GetAdditionalDirectCosts_SheetMetal_PlasmaCut = ADCost0
GetAdditionalDirectCosts_SheetMetal_ProgressiveDie = ADCost1
GetAdditionalDirectCosts_SheetMetal_Shear = ADCost0
GetAdditionalDirectCosts_SheetMetal_StdPress = ADCost0
GetAdditionalDirectCosts_SheetMetal_TandemPress = ADCost0
GetAdditionalDirectCosts_SheetMetal_TransferPress = ADCost0
GetAdditionalDirectCosts_SheetMetal_TurretPress = ADCost0

    ADCost0 = GetAdditionalDirectCosts
    ADCost1 = results.additionalDirectCosts

/*
 * Function : GetAdditionalAmortizedInvestment
 */
GetAdditionalAmortizedInvestment_SheetMetal_3DLaser = AAInvest0
GetAdditionalAmortizedInvestment_SheetMetal_BendBrake = AAInvest0
GetAdditionalAmortizedInvestment_SheetMetal_CTL = AAInvest0
GetAdditionalAmortizedInvestment_SheetMetal_Deburr = AAInvest0
GetAdditionalAmortizedInvestment_SheetMetal_Deslag = AAInvest0
GetAdditionalAmortizedInvestment_SheetMetal_GenericPress = AAInvest0
GetAdditionalAmortizedInvestment_SheetMetal_LaserCut = AAInvest0
GetAdditionalAmortizedInvestment_SheetMetal_MaterialStock = AAInvest0
GetAdditionalAmortizedInvestment_SheetMetal_OxyFuelCut = AAInvest0
GetAdditionalAmortizedInvestment_SheetMetal_PlasmaCut = AAInvest0
GetAdditionalAmortizedInvestment_SheetMetal_ProgressiveDie = AAInvest1
GetAdditionalAmortizedInvestment_SheetMetal_Shear = AAInvest0
GetAdditionalAmortizedInvestment_SheetMetal_StdPress = AAInvest0
GetAdditionalAmortizedInvestment_SheetMetal_TandemPress = AAInvest0
GetAdditionalAmortizedInvestment_SheetMetal_TransferPress = AAInvest0
GetAdditionalAmortizedInvestment_SheetMetal_TurretPress = AAInvest0

    AAInvest0 = GetAdditionalAmortizedInvestment
        AAInvest1= results.additionalAmortizedInvestment
```

```
/*
 * Function : GetExtraCosts
 */
GetExtraCosts_SheetMetal_3DLaser = ECost0
GetExtraCosts_SheetMetal_BendBrake = ECost0
GetExtraCosts_SheetMetal_CTL = ECost0
GetExtraCosts_SheetMetal_Deburr = ECost0
GetExtraCosts_SheetMetal_Deslag = ECost0
GetExtraCosts_SheetMetal_GenericPress = ECost0
GetExtraCosts_SheetMetal_LaserCut = ECost0
GetExtraCosts_SheetMetal_MaterialStock = ECost0
GetExtraCosts_SheetMetal_OxyFuelCut = ECost0
GetExtraCosts_SheetMetal_PlasmaCut = ECost0
GetExtraCosts_SheetMetal_ProgressiveDie = ECost0
GetExtraCosts_SheetMetal_Shear = ECost0
GetExtraCosts_SheetMetal_StdPress = ECost0
GetExtraCosts_SheetMetal_TandemPress = ECost0
GetExtraCosts_SheetMetal_TransferPress = ECost0
GetExtraCosts_SheetMetal_TurretPress = ECost0

    ECost0 = GetExtraCosts

/*
 * TIMES
 *
 * Function : GetCycleTime
 */
GetCycleTime_SheetMetal_3DLaser(processTime, numOperators) = CycleTime0
GetCycleTime_SheetMetal_BendBrake(processTime, numOperators) = CycleTime0
GetCycleTime_SheetMetal_CTL(processTime, numOperators) = CycleTime0
GetCycleTime_SheetMetal_Deburr(processTime, numOperators) = CycleTime0
GetCycleTime_SheetMetal_Deslag(processTime, numOperators) = CycleTime0
GetCycleTime_SheetMetal_GenericPress(processTime, numOperators) = CycleTime0
GetCycleTime_SheetMetal_LaserCut(processTime, numOperators) = CycleTime0
```

GetCycleTime_SheetMetal_MaterialStock(processTime, numOperators) = CycleTime1

GetCycleTime_SheetMetal_OxyFuelCut(processTime, numOperators) = CycleTime0

GetCycleTime_SheetMetal_PlasmaCut(processTime, numOperators) = CycleTime0

GetCycleTime_SheetMetal_ProgressiveDie(processTime, numOperators) = CycleTime0

GetCycleTime_SheetMetal_Shear(processTime, numOperators) = CycleTime0

GetCycleTime_SheetMetal_StdPress(processTime, numOperators) = CycleTime0

GetCycleTime_SheetMetal_TandemPress(processTime, numOperators) = CycleTime0

GetCycleTime_SheetMetal_TransferPress(processTime, numOperators) = CycleTime0

GetCycleTime_SheetMetal_TurretPress(processTime, numOperators) = CycleTime0

CycleTime0 = GetCycleTime(processTime, numOperators)

CycleTime1 = UNUSED

*/**

** Function : GetLaborTime*

**/*

GetLaborTime_SheetMetal_3DLaser(processTime, cycleTime, numOperators, laborTimeStandard) = LaborTime0

GetLaborTime_SheetMetal_BendBrake(processTime, cycleTime, numOperators, laborTimeStandard) = LaborTime0

GetLaborTime_SheetMetal_CTL(processTime, cycleTime, numOperators, laborTimeStandard) = LaborTime0

GetLaborTime_SheetMetal_Deburr(processTime, cycleTime, numOperators, laborTimeStandard) = LaborTime0

GetLaborTime_SheetMetal_Deslag(processTime, cycleTime, numOperators, laborTimeStandard) = LaborTime0

GetLaborTime_SheetMetal_GenericPress(processTime, cycleTime, numOperators, laborTimeStandard) = LaborTime0

GetLaborTime_SheetMetal_LaserCut(processTime, cycleTime, numOperators, laborTimeStandard) = LaborTime0

GetLaborTime_SheetMetal_MaterialStock(processTime, cycleTime, numOperators, laborTimeStandard) = LaborTime1

GetLaborTime_SheetMetal_OxyFuelCut(processTime, cycleTime, numOperators, laborTimeStandard) = LaborTime0

GetLaborTime_SheetMetal_PlasmaCut(processTime, cycleTime, numOperators, laborTimeStandard) = LaborTime0

GetLaborTime_SheetMetal_ProgressiveDie(processTime, cycleTime, numOperators, *laborTimeStandard) = LaborTime0*

GetLaborTime_SheetMetal_Shear(processTime, cycleTime, numOperators, laborTimeStandard) = LaborTime0

GetLaborTime_SheetMetal_StdPress(processTime, cycleTime, numOperators, laborTimeStandard) = LaborTime0

GetLaborTime_SheetMetal_TandemPress(processTime, cycleTime, tandemLineNumOperators, stageCount, laborTimeStandard) = LaborTime2

GetLaborTime_SheetMetal_TransferPress(processTime, cycleTime, numOperators, laborTimeStandard) = LaborTime0

GetLaborTime_SheetMetal_TurretPress(processTime, cycleTime, numOperators, laborTimeStandard) = LaborTime0

LaborTime0 = GetLaborTime(processTime, cycleTime, numOperators, laborTimeStandard)

LaborTime1 = UNUSED

*LaborTime2 = { (cycleTime * laborTimeStandard) if setup.transferMethodTandem == 'Manual'*

*(cycleTime * tandemLineNumOperators)/stageCount otherwise }*

*/**

** Function : GetSetupTimePerPart*

**/*

GetSetupTimePerPart_SheetMetal_3DLaser(batchSize, cycleTime, batchSetupTime) = SetupTime0

GetSetupTimePerPart_SheetMetal_BendBrake(batchSize, cycleTime, batchSetupTime) = SetupTime0

GetSetupTimePerPart_SheetMetal_CTL(batchSize, cycleTime, batchSetupTime) = SetupTime0

GetSetupTimePerPart_SheetMetal_Deburr(batchSize, cycleTime, batchSetupTime) = SetupTime0

GetSetupTimePerPart_SheetMetal_Deslag(batchSize, cycleTime, batchSetupTime) = SetupTime0

GetSetupTimePerPart_SheetMetal_GenericPress(batchSize, cycleTime, batchSetupTime) = SetupTime0

GetSetupTimePerPart_SheetMetal_LaserCut(batchSize, cycleTime, batchSetupTime) = SetupTime0

GetSetupTimePerPart_SheetMetal_MaterialStock(batchSize, cycleTime, batchSetupTime) =

SetupTime0

GetSetupTimePerPart_SheetMetal_OxyFuelCut(batchSize, cycleTime, batchSetupTime) = SetupTime0

GetSetupTimePerPart_SheetMetal_PlasmaCut(batchSize, cycleTime, batchSetupTime) = SetupTime0

GetSetupTimePerPart_SheetMetal_ProgressiveDie(batchSize, cycleTime, batchSetupTime) = SetupTime0

GetSetupTimePerPart_SheetMetal_Shear(batchSize, cycleTime, batchSetupTime) = SetupTime0

GetSetupTimePerPart_SheetMetal_StdPress(batchSize, cycleTime, batchSetupTime) = SetupTime0

GetSetupTimePerPart_SheetMetal_TandemPress(batchSize, cycleTime, batchSetupTime) = SetupTime0

GetSetupTimePerPart_SheetMetal_TransferPress(batchSize, cycleTime, batchSetupTime) = SetupTime0

GetSetupTimePerPart_SheetMetal_TurretPress(batchSize, cycleTime, batchSetupTime) = SetupTime0

SetupTime0 = GetSetupTimePerPart(batchSize, cycleTime, batchSetupTime)

*/**
** Function : GetLaborHandlingTime*
**/*

GetLaborHandlingTime_SheetMetal_3DLaser = LHTime1

GetLaborHandlingTime_SheetMetal_BendBrake = LHTime0

GetLaborHandlingTime_SheetMetal_CTL = LHTime2

GetLaborHandlingTime_SheetMetal_Deburr = LHTime0

GetLaborHandlingTime_SheetMetal_Deslag = LHTime0

GetLaborHandlingTime_SheetMetal_GenericPress = LHTime3

GetLaborHandlingTime_SheetMetal_LaserCut = LHTime0

GetLaborHandlingTime_SheetMetal_MaterialStock = LHTime0

GetLaborHandlingTime_SheetMetal_OxyFuelCut = LHTime0

GetLaborHandlingTime_SheetMetal_PlasmaCut = LHTime0

GetLaborHandlingTime_SheetMetal_ProgressiveDie = LHTime0

GetLaborHandlingTime_SheetMetal_Shear = LHTime4

GetLaborHandlingTime_SheetMetal_StdPress(numDieOps, laborHandlingAllowance) = LHTime5

GetLaborHandlingTime_SheetMetal_TandemPress = LHTime6

GetLaborHandlingTime_SheetMetal_TransferPress = LHTime0

GetLaborHandlingTime_SheetMetal_TurretPress = LHTime7

LHTime0 = GetLaborHandlingTime

LHTime1 = select first(entry.allowance) from sm3DLaserHandling entry where entry.weight >=finishMass order by entry.weight

LHTime2 = select first(entry.allowance) from smCTLHandling entry where entry.weight >=blankMass order by entry.weight

LHTime3 = select first(entry.allowance) from smGenericPressHandling as entry where entry.weight >= partMass order by entry.weight

LHTime4 = select first(entry.allowance) from smShearHandling entry where entry.weight >= blankMass order by entry.weight

*LHTime5 = numDieOps * laborHandlingAllowance*

LHTime6 = select first(entry.allowance) from smGenericPressHandling as entry where entry.weight >= partMass order by entry.weight

LHTime7 = select first(entry.allowance) from smTurretPressHandling as entry where entry.weight >= partMass order by entry.weight

```
/*
 * PERIOD COSTS
 *
 * Function : GetPeriodOverhead
 */
GetPeriodOverhead_SheetMetal_3DLaser(periodOverheadCoefficient) = POH0
GetPeriodOverhead_SheetMetal_BendBrake(periodOverheadCoefficient) = POH0
GetPeriodOverhead_SheetMetal_CTL(periodOverheadCoefficient) = POH0
GetPeriodOverhead_SheetMetal_Deburr(periodOverheadCoefficient) = POH0
GetPeriodOverhead_SheetMetal_Deslag(periodOverheadCoefficient) = POH0
GetPeriodOverhead_SheetMetal_GenericPress(periodOverheadCoefficient) = POH0
GetPeriodOverhead_SheetMetal_LaserCut(periodOverheadCoefficient) = POH0
GetPeriodOverhead_SheetMetal_MaterialStock(periodOverheadCoefficient) = POH0
GetPeriodOverhead_SheetMetal_OxyFuelCut(periodOverheadCoefficient) = POH0
GetPeriodOverhead_SheetMetal_PlasmaCut(periodOverheadCoefficient) = POH0
GetPeriodOverhead_SheetMetal_ProgressiveDie(periodOverheadCoefficient) = POH0
GetPeriodOverhead_SheetMetal_Shear(periodOverheadCoefficient) = POH0
GetPeriodOverhead_SheetMetal_StdPress(periodOverheadCoefficient) = POH0
```

GetPeriodOverhead_SheetMetal_TandemPress(periodOverheadCoefficient) = POH0

GetPeriodOverhead_SheetMetal_TransferPress(periodOverheadCoefficient) = POH0

GetPeriodOverhead_SheetMetal_TurretPress(periodOverheadCoefficient) = POH0

POH0 = GetPeriodOverhead(periodOverheadCoefficient)

APPENDIX B

B.1 RLW Cost Taxonomy

```
/***************************************
COSTS PER PART
****************************************/
// Direct Costs
  materialCost = results.materialCost
  laborCost = results.laborCost
  directOverheadCost = results.directOverheadCost

/// Other Direct Costs
    expendableToolingCostPerPart = results.expendableToolingCostPerPart
    logisticsCost = results.logisticsCost
    setupCostPerPart = results.setupCostPerPart
    additionalDirectCosts = results.additionalDirectCosts

// Piece Cost

// Fixed Costs
  additionalAmortizedInvestment = results.additionalAmortizedInvestment

// TOTAL

// Fully Burdened Cost
  periodOverhead = results.periodOverhead

// Production Volume
totalProductionVolume = part.annualVolume * part.productionLife

/******************************************
SETUP AND INVESTMENT VALUES
*******************************************/
// Capital Investment
  hardToolingCost = results.hardToolingCost
  fixtureCost = results.fixtureCost
  programmingCost = results.programmingCost
```

```
/****************************************
ADDITIONAL REPORTED VALUES
*****************************************/
// (Material) Utilization
utilization = results.utilization
roughMass = 0
scrapMass = 0

finishMass = {  componentMass + results.finishMass if (op.isRoot)  // only include component mass in the sum at the root level
        results.finishMass otherwise } // includes any additional weld mass

componentMass = select sum(c.weight) from part.subcomponents c

// Times
cycleTime = results.cycleTime
laborTime = results.laborTime
laborHandlingTime = results.laborHandlingTime
```

B.2 GENERAL CONSTANTS

```
// Note: Process specific constants should be stored as site., plant., or lookup table constants

// Fail Message begin/ending text
FLT = '---> CSL ASSERT FAILED: ' // left delimiter
FRT = '<---.  ' // right delimiter
OLT = 'OPTION OFF: ' // left
// Unit conversions
CUBIC_MM_PER_CUBIC_M = 1E9
MICRONS_PER_MM = 1000
MM_PER_M = 1000
MM_PER_IN = 25.4
PI = constants.pi
DEGREES_PER_RADIAN = 180 / constants.pi
SEC_PER_HR = 3600
SEC_PER_MIN = 60
MIN_PER_HR = 60
```

```
// Flags
UNUSED = 0
TODO = 0
UNCONSTRAINED = -1
UNBOUNDED = -1
```

APPENDIX C

C.1 RLW Component Template

Assembly/Machining ::= Assembly ['Other Secondary Processes']

Assembly ::= ['Prepare Dimples':Assembly] 'Pick and Place':Assembly ['Weld Prep':Assembly] Welding ['Weld Clean Up':Assembly]

Welding ::= ['Manual MIG Welding':Assembly] ['Robotic MIG Welding':Assembly] ['Manual Spot Welding':Assembly] ['Remote Laser Welding':Assembly] ['Robotic Spot Welding':Assembly]

RemoteLaserWelding::= SystemLevelOperations

SystemLevelOperations::= ST110 CombiActivity1 ST140 ST150 ST160 ST170 ST180

ST110::= Load3Parts PushButtonToTurnTable Turn180 MoveToGunStand PickUpGun MoveToST110 CreateGeo4WS WalkToST110 UnloadHaloSubAssy

CombiActivity1::= ST120 | ST100

ST120::= ST120A | ST120B

ST120A::= WalkToST120WithHaloSubAssy LoadHaloSubAssyAnd2Parts PushButton WalkToST110

ST120B::= MoveToST120 PickHaloSubAssyAnd2Parts MoveToST140

ST100::= ST100A | ST100B

ST100A::= WalkToST100 LoadDoorInner PushButton

ST100B::= MoveToGunStand DropGun MOveToGripperStand PickUpDoubleGripper MoveToST100 PickDoorInner MoveToSt140

ST140::= ST140A | ST140B

ST140A::= OPenWldFixt PickWldDrFromWldFixt Place3PartInWldFixt ClsWldFixt

ST140B::= OpenDimpFixt PickDimpDrInnerIntoDimpFixt LoadDrInnerIntoDimplFixt MoveToWldWldFixtPositn LoadDimpInnerIntoWldFixt CloseDimplFixt OP140C

ST140C::= TurnTable180 RmtLsrWldDrAssy Dimpl DrInner TurnTable180

ST150::= MoveToST150 DropWldDrAtST150 MoveToST150 Respot_5WS MoveToST150 PickWldDrFrmST150

ST160::= MoveToST160 Cone

ST170::= MoveToST170 Seal

ST180::= MoveToST180 DropCompltdDr

C.2 Cycle Time CSL

// AUTO-GENERATED ON Tuesday, 26 April 2016

GetCycleTime_Assembly_RemoteLaserWelding(processTime, laborTime, numOperations) = TotalCycleTime

*Total_CycleTime = (operationTime * numOperations)*
*OperationTime = (select sum(op.cycleTime) from childOps op) * plant.cycleTimeAdjustmentFactor*
numOperations = 1

GetLaborTime_Assembly_RemoteLaserWelding(processTime,numOperators,laborTimeStandard) = LaborTime1
*LaborTime1 = weldTime * numOperators * laborTimeStandard*

GetLaborTime_Assembly_WeldPrep(processTime, numOperators, laborTimeStandard) = LaborTime2
*LaborTime2 = processTime * numOperators * laborTimeStandard*

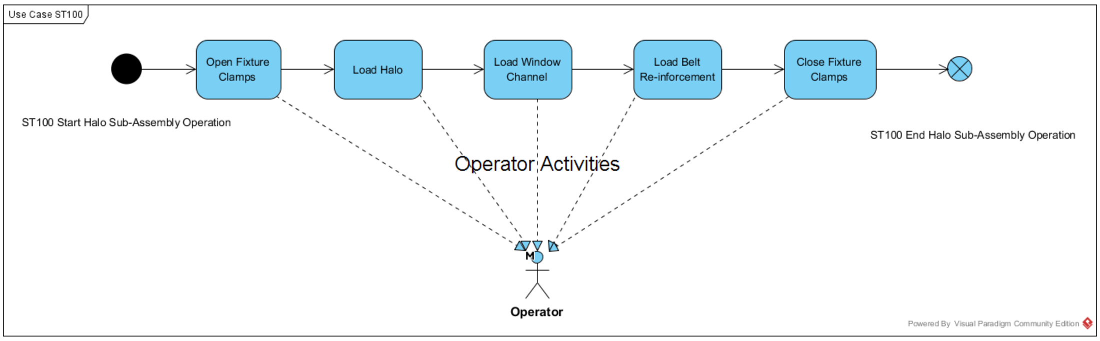
GetLaborTime_Assembly_PickAndPlace(processTime,numOperators, unloadAsmFixtureTime, laborTimeStandard) = LaborTime3
*LaborTime3 = ((select sum(op.cycleTime) from childOps op) + unloadAsmFixtureTime) * numOperators * laborTimeStandard*

APPENDIX D

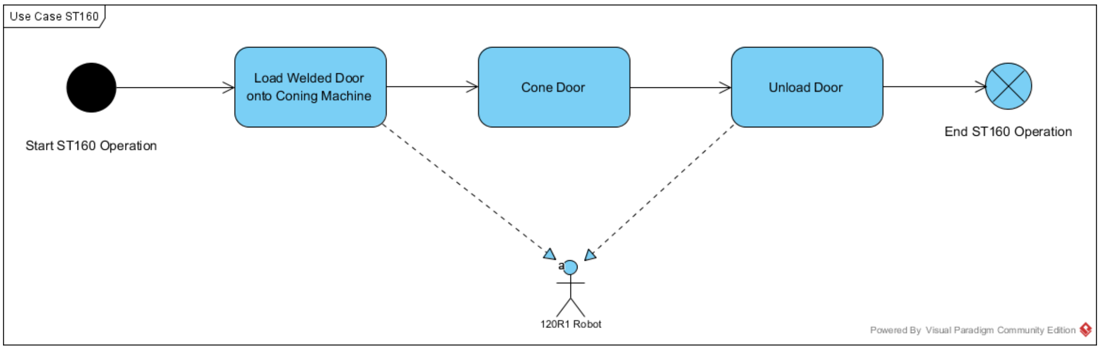
D.1 Complete XML files for Product-Process-Resource models.

The complete models in XML extension can be found in the link below:
https://drive.google.com/file/d/1RwEvN4tlalcapv17DXyhDZEv7zNd2gtA/view?usp=sharing

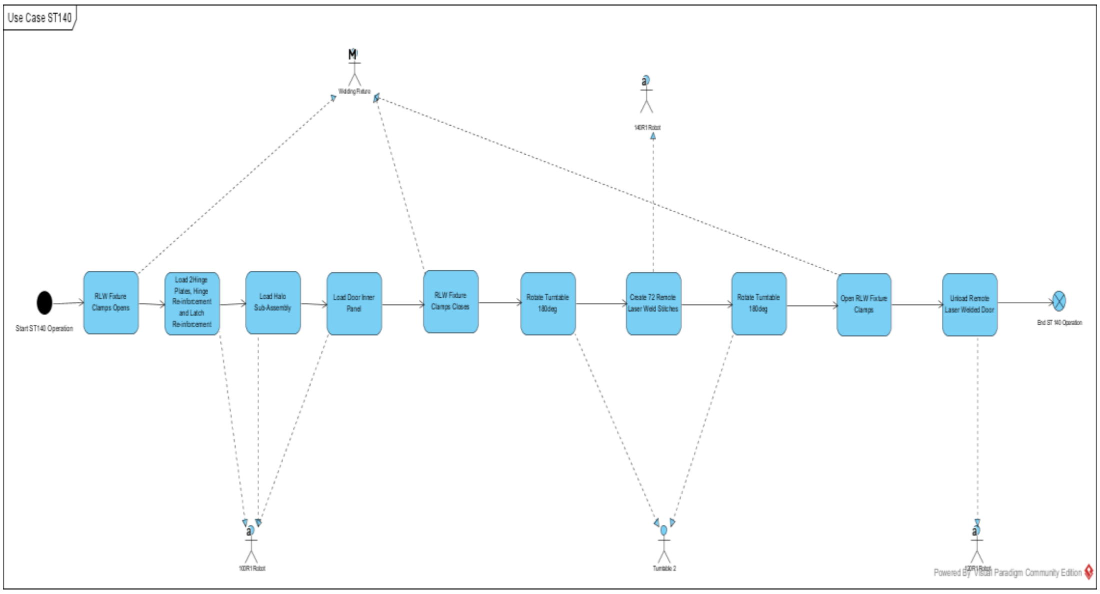
D.2 Use case for ST100



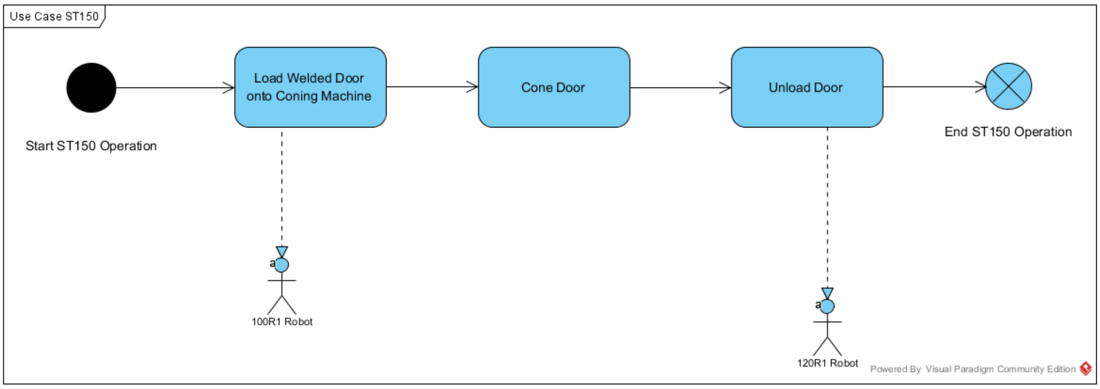
D.3 Use case for ST160



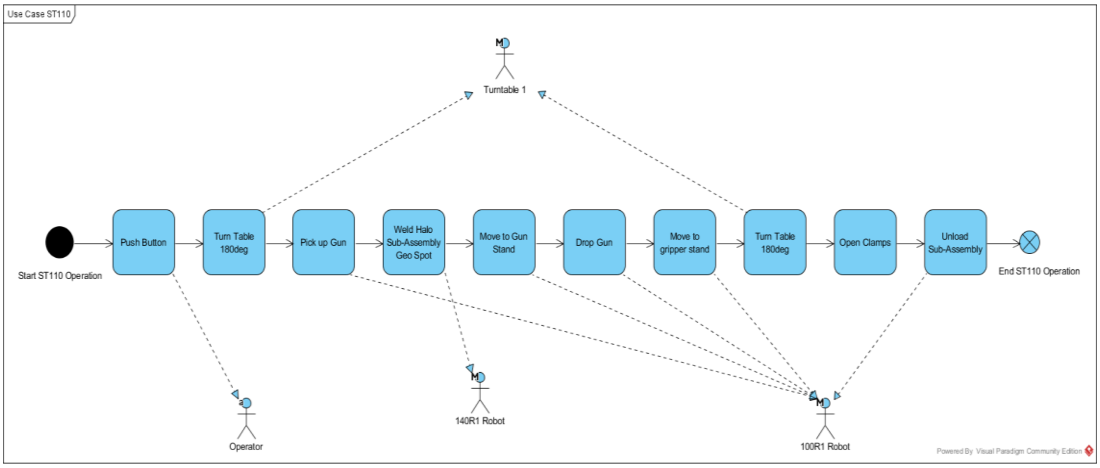
D.4 Use case for ST140



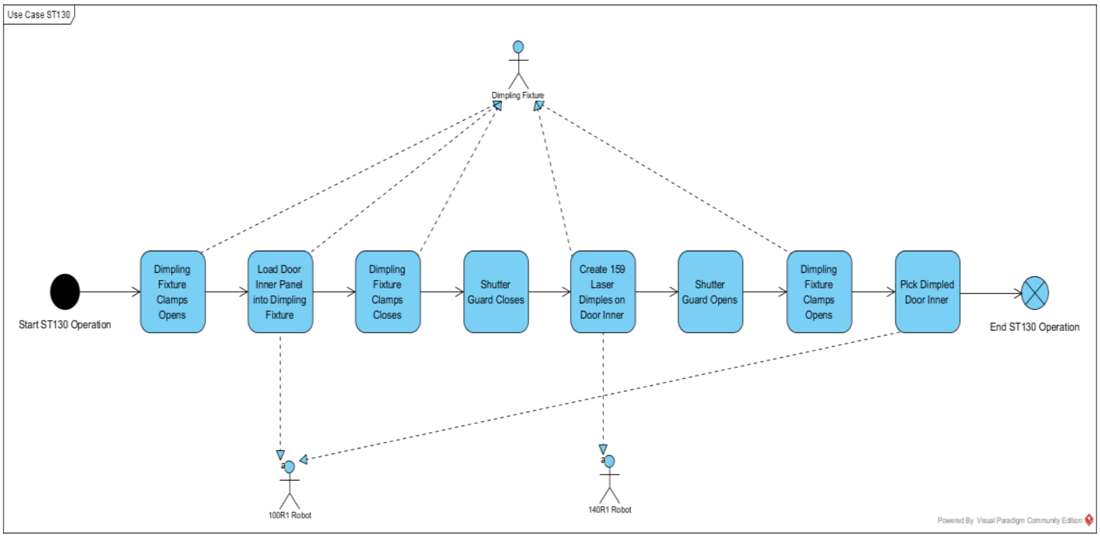
D.5 Use case for ST150



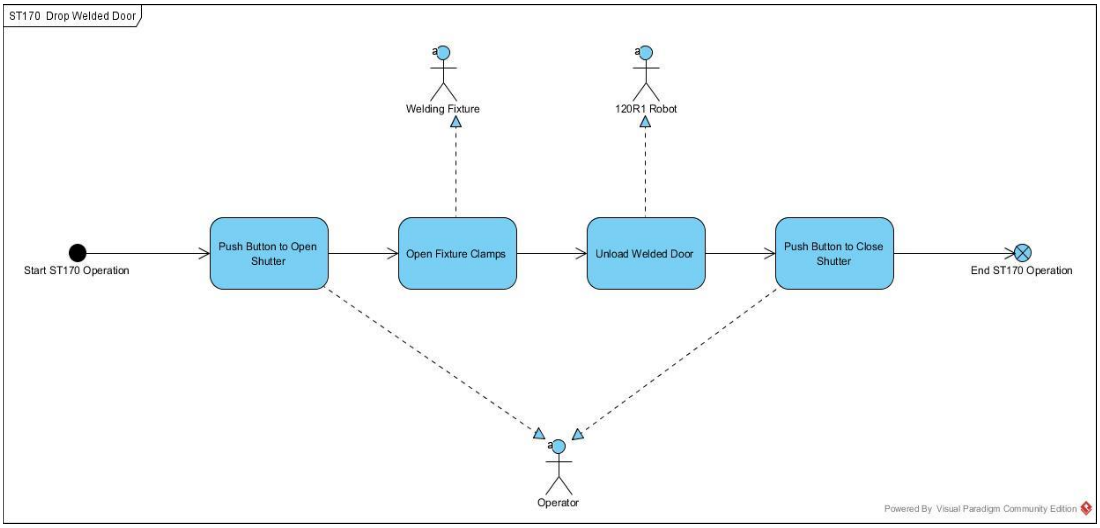
D.6 Use case for ST110



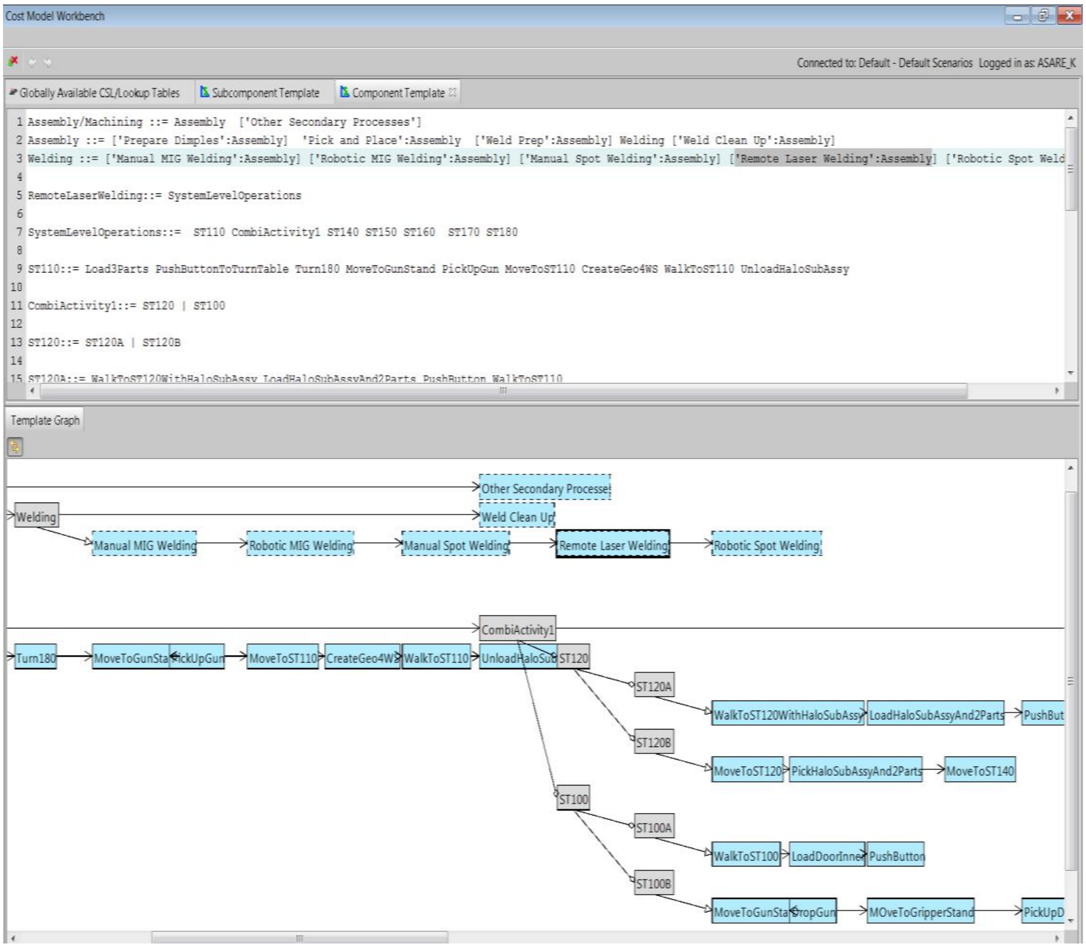
D.7 Use case for ST130



D.8 Use case for ST170



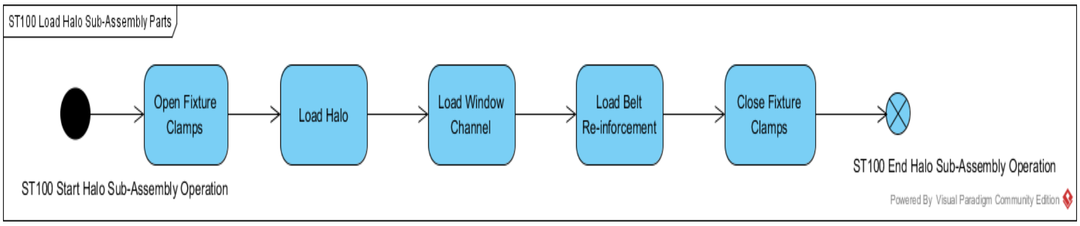
D.9 aPriori's Cost Model Workbench



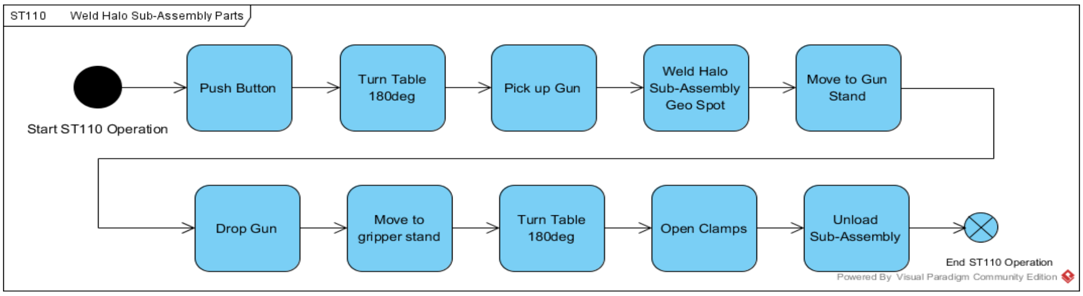
APPENDIX E

Workstations Flow Models

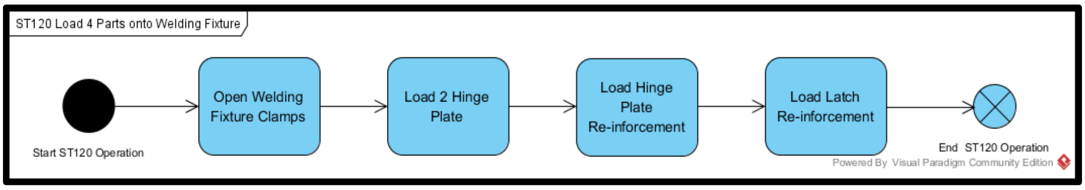
E.1



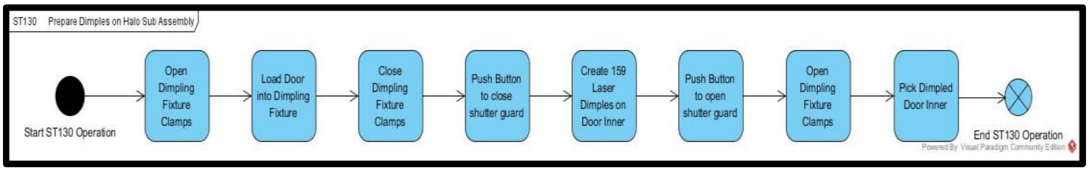
E.2



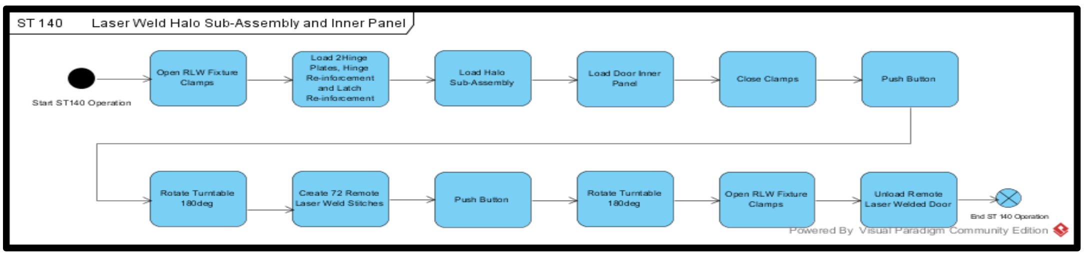
E.3



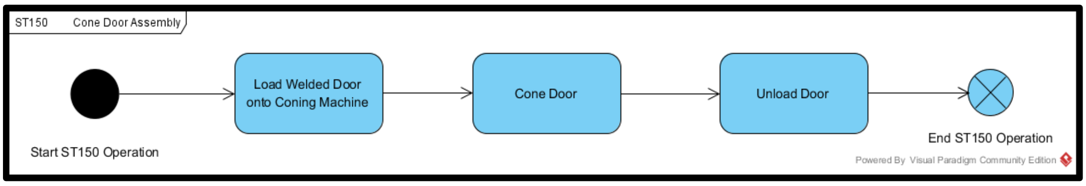
E.4



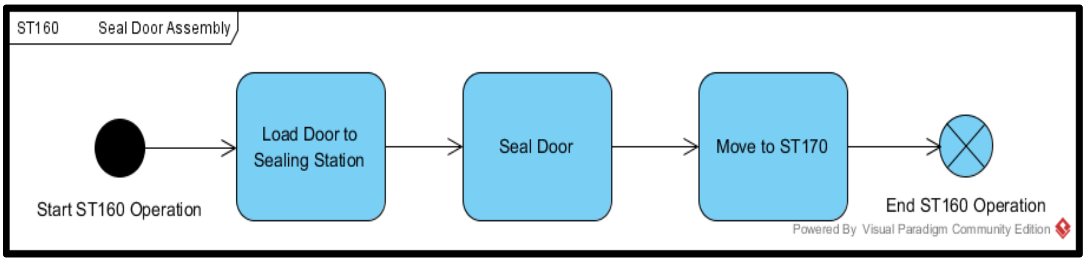
E.5



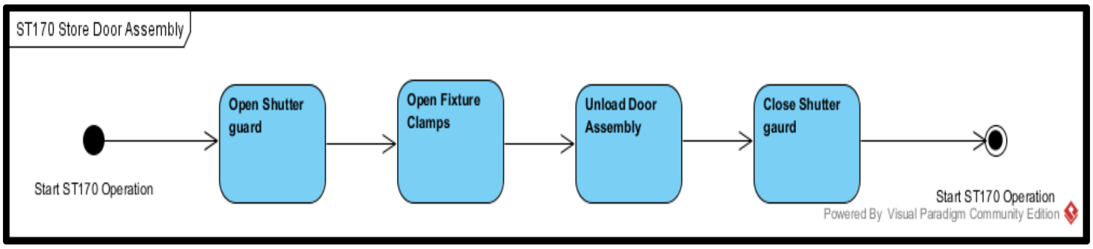
E.6


ST150 Cone Door Assembly
Load Welded Door onto Coning Machine
Cone Door
Unload Door
Start ST150 Operation
End ST150 Operation
Powered By Visual Paradigm Community Edition

E.7


ST160 Seal Door Assembly
Load Door to Sealing Station
Seal Door
Move to ST170
Start ST160 Operation
End ST160 Operation
Powered By Visual Paradigm Community Edition

E.8

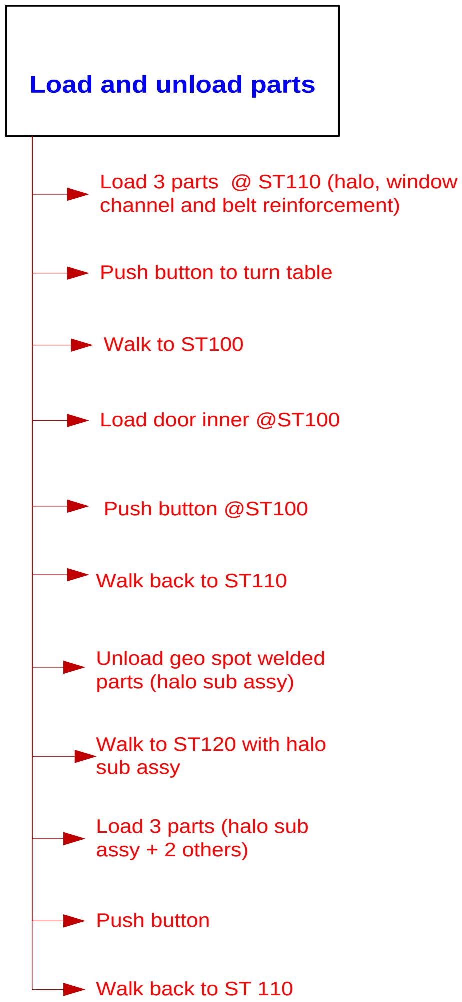

ST170 Store Door Assembly
Open Shutter guard
Open Fixture Clamps
Unload Door Assembly
Close Shutter gaurd
Start ST170 Operation
Start ST170 Operation
Powered By Visual Paradigm Community Edition

APPENDIX F

Resource Models

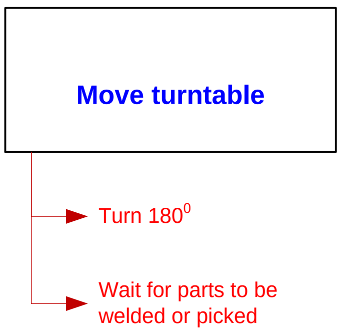
F.1

Operator activities



F.2

Turntable 1 activities

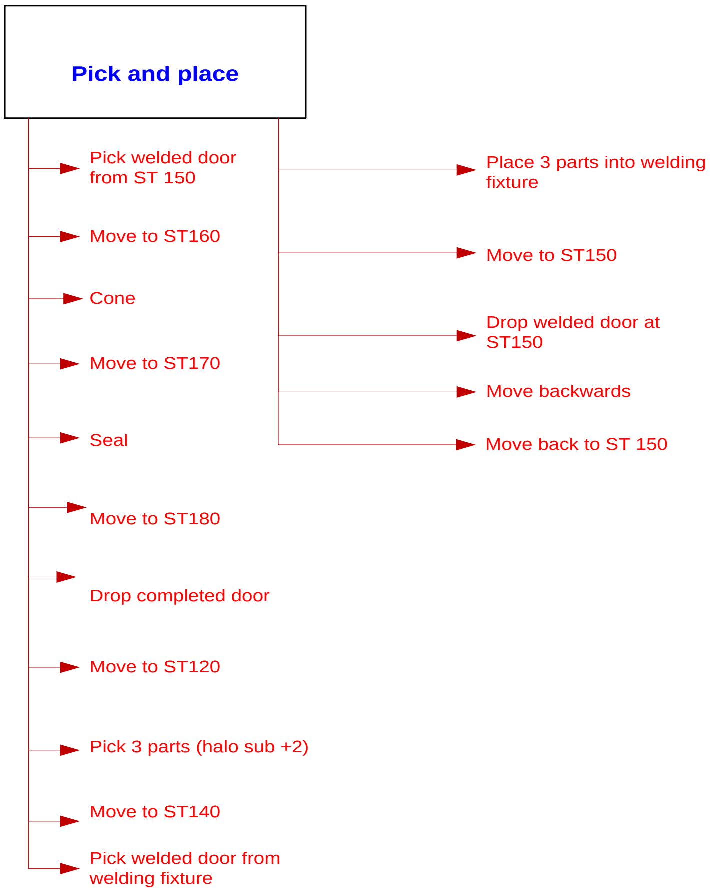


F.3


100R1 activities
Pick and weld
Move from ST140 to gripper stand
Drop gripper
Move to gun stand
Pick up gun
Move to ST150
Respot (5?)
Move to ST110
Create geometry spot (4?)
Move to gun stand
Drop gun
Move to gripper stand
Pick up double gripper
Move to ST100
Pick door inner
Move to ST140
Pick dimpled door inner
Load door inner into dimpling fixture
Move to welding fixture postion
Load dimpled inner into welding fixture

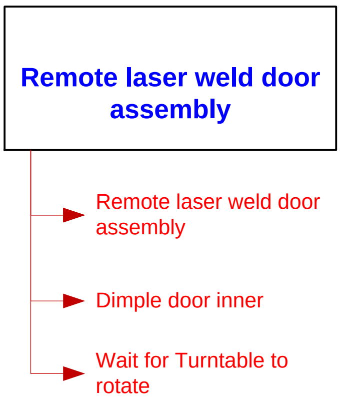
F.4

120R1Robot activities

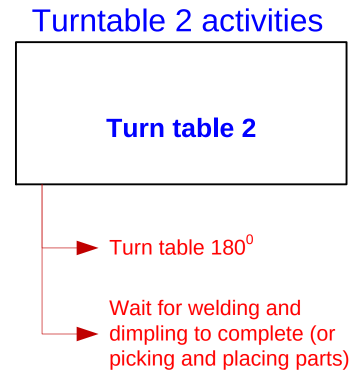


F.5

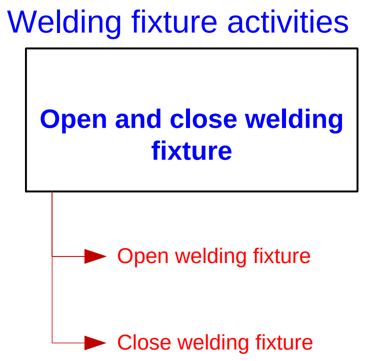
140R1 RLW activities



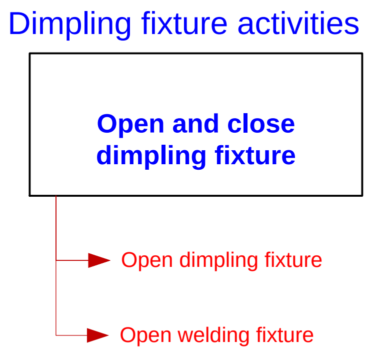
F.6


Turntable 2 activities
Turn table 2
Turn table 180^0
Wait for welding and dimpling to complete (or picking and placing parts)

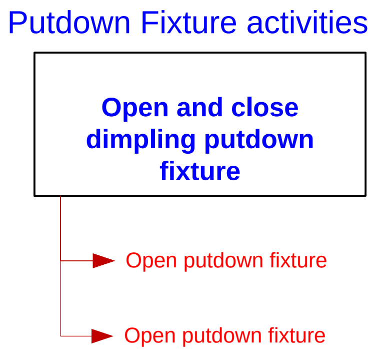
F.7


Welding fixture activities
Open and close welding fixture
Open welding fixture
Close welding fixture

F.8

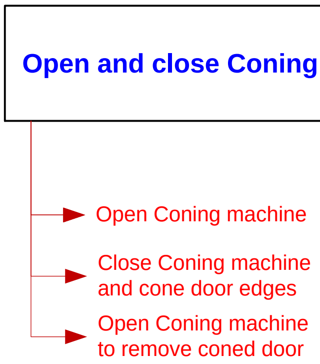

Dimpling fixture activities
Open and close dimpling fixture
Open dimpling fixture
Open welding fixture

F.9


Putdown Fixture activities
Open and close dimpling putdown fixture
Open putdown fixture
Open putdown fixture

F.10

Coning Machine activities



F.11

Sealing Machine activities